future internet

MDPI

Article

Cross-Gen: An Efficient Generator Network for Adversarial Attacks on Cross-Modal Hashing Retrieval

Chao Hu [1], Li Chen [1], Sisheng Li [1], Yin Yi [1], Yu Zhan [2], Chengguang Liu [3], Jianling Liu [4,*] and Ronghua Shi [1]

[1] School of Electronic Information, Central South University, Changsha 410083, China; huchao@csu.edu.cn (C.H.); 194702040@csu.edu.cn (L.C.); lisisheng@csu.edu.cn (S.L.); yiyin78@csu.edu.cn (Y.Y.)
[2] China Telecom, Changsha 410083, China; zhany3@chinatelecom.cn
[3] Big Data Institute, Central South University, Changsha 410083, China; liuchengguang@csu.edu.cn
[4] State Key Laboratory of Powder Metallurgy, Central South University, Changsha 410083, China
* Correspondence: jianling.liu@csu.edu.cn

Abstract

Research on deep neural network (DNN)-based multi-dimensional data visualization has thoroughly explored cross-modal hash retrieval (CMHR) systems, yet their vulnerability to malicious adversarial examples remains evident. Recent work improves the robustness of CMHR networks by augmenting training datasets with adversarial examples. Prior approaches typically formulate the generation of cross-modal adversarial examples as an optimization problem solved through iterative methods. Although effective, such techniques often suffer from slow generation speed, limiting research efficiency. To address this, we propose a generative-based method that enables rapid synthesis of adversarial examples via a carefully designed adversarial generator network. Specifically, we introduce Cross-Gen, a parallel cross-modal framework that constructs semantic triplet data by interacting with the target model through query-based feedback. The generator is optimized using a tailored objective comprising adversarial loss, reconstruction loss, and quantization loss. The experimental results show that Cross-Gen generates adversarial examples significantly faster than iterative methods while achieving competitive attack performance.

Keywords: adversarial attack; cross-modal Hamming retrieval; generative-based network; deep hashing

Academic Editors: Wei Zong and Yang-Wai Chow

Received: 23 October 2025
Revised: 30 November 2025
Accepted: 11 December 2025
Published: 13 December 2025

Citation: Hu, C.; Chen, L.; Li, S.; Yi, Y.; Zhan, Y.; Liu, C.; Liu, J.; Shi, R. Cross-Gen: An Efficient Generator Network for Adversarial Attacks on Cross-Modal Hashing Retrieval. *Future Internet* **2025**, *17*, 573. https://doi.org/10.3390/fi17120573

1. Introduction

The exploration of multi-dimensional data visualization in deep learning has garnered significant attention due to its interpretability [1,2]. However, applying such techniques to cross-modal hash retrieval (CMHR) systems remains challenging as heterogeneous modalities—such as images and text—exhibit distinct representation formats and require modality-specific feature extraction methods. A typical CMHR system allows users to submit a query in one modality (e.g., an image or text) and retrieve semantically relevant results from the other modality. To bridge this gap, recent work widely employs deep learning approaches [3–5] for cross-modal feature embedding.

Despite their effectiveness, deep neural network (DNN)-based models are inherently vulnerable to imperceptible adversarial perturbations [6–8]. Seminal work by Szegedy et al. [9] demonstrated that minor input perturbations can induce substantial output changes due to the highly nonlinear nature of DNNs. This vulnerability enables adversaries to manipulate retrieval results, posing serious security risks. Critically, such perturbations

are often imperceptible to humans. For instance, image perturbations involve subtle pixel modifications, while text perturbations may include minor alterations such as reordering words, inserting punctuation, or replacing characters.

Adversarial attacks are commonly categorized based on the adversary's knowledge of the target model: white-box attacks [10–12], where full access to model parameters and architecture is assumed, and black-box attacks [13,14], where such information is unavailable. In practice, black-box settings are more realistic but also more challenging. Black-box attacks are further divided into query-based and transfer-based approaches. Query-based methods [15,16] estimate gradients through numerous queries to the target model, while transfer-based methods [17] generate perturbations using a surrogate model and exploit transferability to attack the target. Both paradigms, however, incur high computational costs or require extensive querying, highlighting the need for efficient general-purpose methods for rapid adversarial example generation.

While extensive research has addressed adversarial attacks in single-modality systems—such as image classification [18] and machine translation [19]—the cross-modal retrieval setting remains underexplored. The existing attack strategies generally fall into two categories: iterative-based [9] and generative-based methods [20,21]. Iterative approaches formulate the attack as an optimization problem solved via gradient descent, progressively refining perturbations toward an optimal solution. However, they demand repeated gradient computations or backward passes through the target model, resulting in high time overhead. In contrast, generative-based methods [21,22] leverage trained generator networks to produce adversarial examples efficiently, showing promise for scalable attacks in deep hash-based CMHR systems. For example, Wang et al. [21] employed a generator–discriminator architecture trained with target labels encoded as semantic guidance within a self-reconstruction framework.

In this work, we focus on the white-box setting and propose Cross-Gen, a parallel generator framework comprising dedicated image and text generators, designed to learn a general adversarial mapping that bypasses costly instance-specific optimization. Cross-Gen aims to approximate and invert the decision boundary of the target CMHR model—i.e., to generate inputs that cause the model to retrieve semantically dissimilar results. During training, we first collect cross-modal data pairs by querying the target model. The original samples are then fed into Cross-Gen to produce adversarial counterparts, which are evaluated by the target model to compute loss signals for end-to-end parameter updates. Specifically, we design three complementary loss functions: (1) an adversarial loss that minimizes the Hamming distance between hash codes of adversarial and target (dissimilar) samples; (2) a quantization loss that reduces the discrepancy between continuous outputs and binary hash codes; and (3) a reconstruction loss to ensure that the generated examples remain visually or semantically close to the originals. The entire framework is optimized using the Adam optimizer [23]. Once trained, Cross-Gen enables rapid generation of effective adversarial examples without per-instance optimization.

We evaluate our approach on two standard benchmarks—FLICKR-25K [24] and NUS-WIDE [25]—across multiple backbone architectures. The experimental results demonstrate that Cross-Gen generates high-quality adversarial examples significantly faster than iterative methods while achieving competitive attack performance.

In summary, our contributions are as follows:

- We propose Cross-Gen, an efficient parallel generator framework that rapidly learns the decision behavior of a target CMHR model and generates cross-modal adversarial examples at inference time without iterative optimization.

- We design a training strategy that acquires relevant cross-modal pairs via interaction with the target model and optimizes the generator using a composite objective combining adversarial, quantization, and reconstruction losses.
- We conduct comprehensive experiments on FLICKR-25K and NUS-WIDE with diverse model backbones, demonstrating that Cross-Gen outperforms iterative baselines in generation speed while maintaining strong attack efficacy.

The rest of this paper is organized as follows. Section 2 provides background on cross-modal retrieval systems. Section 3 formulates the attack problem against CMHR systems and presents our Cross-Gen solution. Section 4 describes the experimental setup—including the datasets, implementation details, results, and analysis. Section 5 reviews the related work on CMHR and adversarial attacks, covering both iterative- and generative-based approaches. Finally, Section 6 concludes the paper and outlines directions for future work.

2. Background

In this section, we detail the composition and training process of CMHR and systematically describe its adversarial attack formulation.

2.1. Cross-Modal Hash Retrieval System

Compared to traditional feature extraction methods—such as spectral hashing [26], color histograms [27], and bag-of-words (BoW) models [28]—recent deep neural network (DNN)-based cross-modal hash retrieval (CMHR) systems [3–5,29] have gained prominence due to their powerful data representation capabilities. When presented with cross-modal inputs such as images or text, a DNN-based CMHR system encodes them into feature vectors using dedicated neural networks and retrieves relevant results based on vector similarity. This paradigm eliminates the need for manual feature engineering; instead, optimal network parameters are learned through large-scale training, enabling high-performance retrieval in an end-to-end manner.

Deep hashing has become a key enabler of efficient CMHR, particularly when combined with approximate nearest neighbor (ANN) search [30], which offers low storage requirements and fast retrieval. ANN-based hashing maps high-dimensional features into compact binary codes by optimizing semantic similarity in the hash space. Through this process, well-trained DNNs learn to preserve semantic relationships across modalities—e.g., aligning image and text representations—thereby establishing cross-modal correlations. A typical DNN-based cross-modal retrieval framework is illustrated in Figure 1.

CMHR systems rely on a shared latent space to generate compact hash codes from heterogeneous modalities. The existing DNN-based approaches fall broadly into two categories: unsupervised frameworks [31,32] and supervised frameworks [33–35]. Supervised methods leverage labeled data to bridge the semantic gap and achieve higher retrieval accuracy, whereas unsupervised methods, although more flexible in label-scarce scenarios, require sophisticated designs to capture high-level semantic structures.

The training of a CMHR model typically follows three sequential steps:

- Embedding semantically aligned cross-modal data pairs into a joint vector space.
- Designing an optimization objective that enhances both semantic consistency and Hamming-space similarity between paired samples.
- Generating binary hash codes for all cross-modal data and storing them in a database to support efficient retrieval.

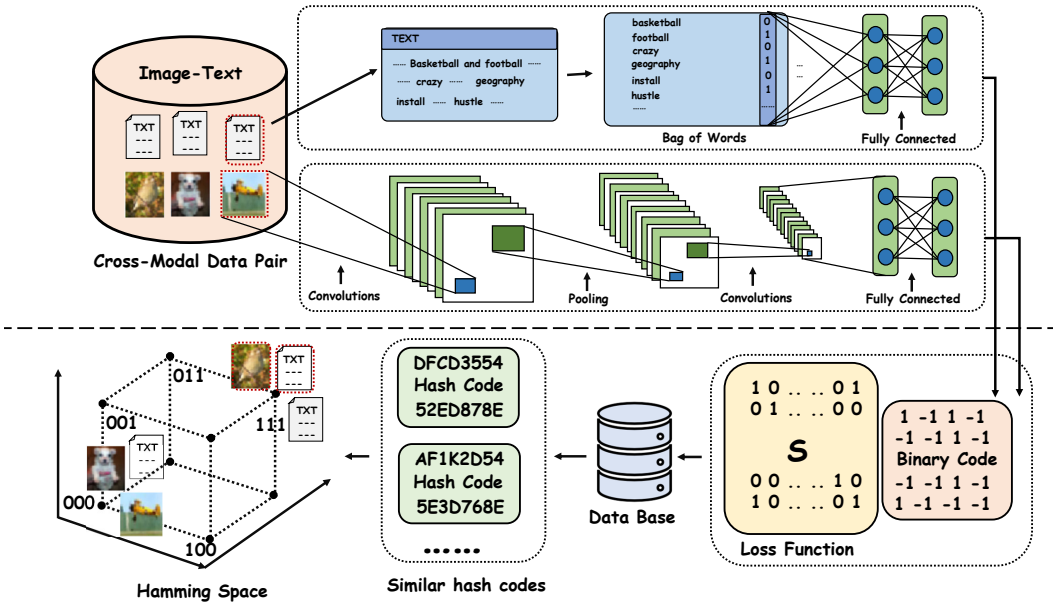

Figure 1. Illustration of the cross-modal retrieval network.

Given a cross-modal dataset with M data points $O = \{o_i\}_i^M$ and $o_i = \{o_i^v, o_i^t\}$, where o^v and o^t represent image and text modality data with similar semantics, the goal of the deep cross-modal hash retrieval model is to construct two efficient functions $\mathcal{F}^*(\cdot)$ and $* \in \{v, t\}$, which project the original data points of two modality types, o^v and o^t, into shared hash space. The cross-modal data point can be represented as a binary-like representation H^*, which is the output of the target model. Hence, CMHR outputs the hash code B^* by the `sign` function, and the above process can be formally described as

$$H^* = \mathcal{F}^*(o^*), B^* = \texttt{sign}(H^*), H^* \in [-1, 1]^K \tag{1}$$

where K is the hash code length, and $* \in \{v, t\}$ denote the image and text data modality.

2.2. Problem Formulation

Suppose a well-trained CMHR model $\mathcal{F}(\cdot)$ can efficiently transfer two similar semantic modal data into a related hash space. For instance, the CMHR model encourages that the hash code of the query data point o_A^v is similar to the positive text data point o_P^t with semantic content. On the contrary, promote the hash code of the negative text data point o_N^t with o_A^v dissimilar. Hence, the CMHR model aims to obtain the optimal model parameter δ_F to retain the semantic relationship of cross-modal data, which can be formulated as

$$\underset{\delta_F}{argmin}\ \texttt{Ham}(\mathcal{F}^v(o_A^v), \mathcal{F}^t(o_P^t)) - \texttt{Ham}(\mathcal{F}^v(o_A^v), \mathcal{F}^t(o_N^t)) + \varphi \tag{2}$$

φ is the margin constant and generally set as $\frac{K}{2}$. $\texttt{Ham}(\cdot, \cdot)$ is the Hamming distance function to measure the similarity between hash codes; i.e., $\texttt{Ham} = \frac{1}{2}(K - < V, T >)$, where $< \cdot, \cdot >$ is the dot product function. Moreover, the generated adversarial examples are bounded by the projection function, which aims to encourage visual imperceptibility. We formulated it as

$$o_A^{'} = \texttt{CLIP}(\texttt{min}(o_A + \epsilon, \texttt{max}(\mathcal{G}(o_A), o_A - \epsilon))) \tag{3}$$

where ϵ is the constant to ensure the imperceptibility of adversarial perturbations. Finally, given the image–text triplet instances $\{o_A^v, o_P^t, o_N^t\}$, we aim to create imperceptible adver-

sarial perturbation to damage the relationship between the triplet cross-modal data, finally degrading the performance of the CMHR model.

3. Framework

In this section, we first introduce the framework overview of our Cross-Gen on adversarial attacks on CMHR, and then we present the details of the proposed Cross-Gen.

3.1. Overview

We present the complete framework of our proposed Cross-Gen in Figure 2. First, we construct triplet data pairs—comprising anchor, positive, and negative samples—by querying the target model. Next, we introduce two parallel generator networks: one for adversarial image generation and another for adversarial text generation. Specifically, the original image is fed into the adversarial image generator to produce a perturbed image, which is then passed through an image discriminator to generate hash codes. An analogous pipeline is employed for text: the original text input is processed by the adversarial text generator, and the resulting output is used to derive hash codes. Finally, we design three loss functions to guide training: reconstruction loss, adversarial loss, and quantization loss.

In the following sub-section, we present the steps of the proposed framework in detail.

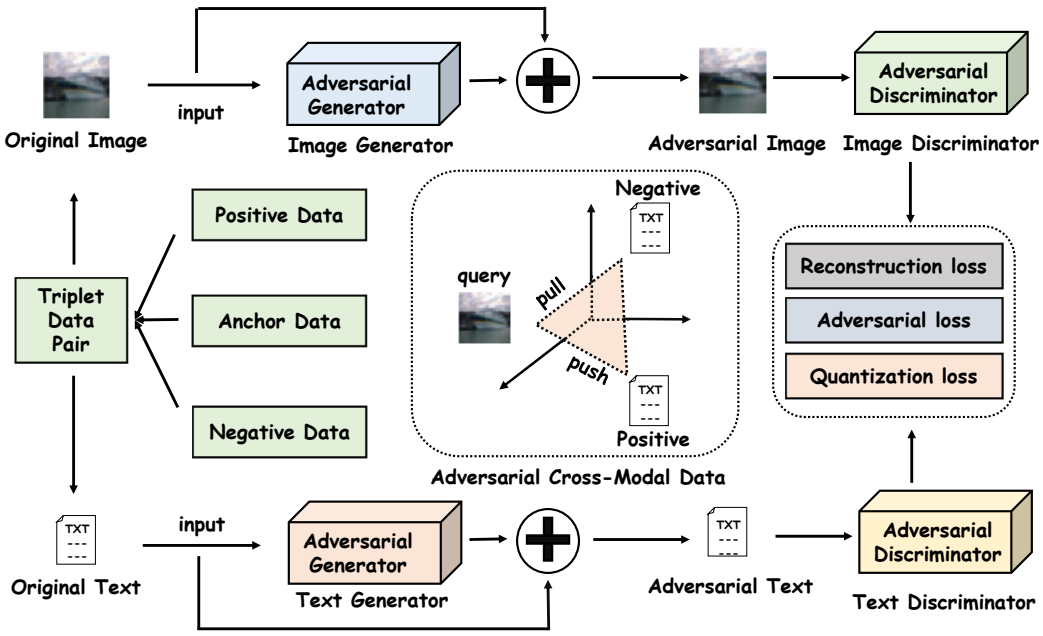

Figure 2. The framework illustrates our parallel generator network (Cross-Gen) (i.e., adversarial image generator and adversarial text generator).

3.2. Cross-Modal Data Adversarial Generator

As mentioned above, since the adversarial networks are subject to the label input for training and finally obtain attack effects (i.e., unsupervised scenarios), the challenge is to effectively promote more scenarios. This paper mainly explores the cross-modal generator network (Cross-Gen) in unsupervised scenarios (i.e., without data labels). We are inspired by the advanced research [36] to maximize the distance between original and adversarial examples in the hash space. Hence, the Cross-Gen network we proposed needs to achieve three goals:

- Adversarial and original input data have significant Hamming distance.
- In the training process, the network needs to generate compact hash codes.
- The generated adversarial examples are invisible from the original input examples.

The Cross-Gen framework is mainly composed of adversarial images, text generators, and discriminator (this paper sets the target model as the discriminator). Cross-Gen aims to learn the target model's decision knowledge and conducts adversarial attacks by combining a two-stream cross-modal retrieval network (i.e., image and text generator) of images and text. Cross-Gen includes the adversarial generator $\mathcal{G}_{\delta_g}$ and discriminator $\mathcal{D}_{\delta_d}$. δ_g and δ_d are the network parameters of the adversarial generator and discriminator, respectively. Among them, we focus on optimizing the adversarial generator $\mathcal{G}$ with its parameter δ_g and setting the target model as the adversarial discriminator $\mathcal{D}_{\delta_d}$. Our designed framework aims to make the adversarial generator $\mathcal{G}$ learn the decision knowledge through the distribution of cross-modal data points returned by the target model. In many common scenarios, researchers cannot easily obtain the data labels. Hence, we attempt to acquire its decision knowledge by communicating with the target model. Specifically, by inputting cross-modal data $o^v(o^t)$, we obtain a cross-modal query set of size Q and the corresponding cross-modal database $S = o^v, o^{t\,Q}$ returned by the target model based on its internally designed hash space. Continuously, we can generate cross-modal triples for every query data point $o_A^v(o_A^t)$, which include positive data points $o_P^t(o_P^v)$ and negative data points $o_N^t(o_N^v)$, i.e., $\{o_A^v, o_P^t, o_N^t\}, \{o_A^t, o_P^v, o_N^v\}$. The smaller Hamming distances of hash codes of data points $o_A^v(o_A^t)$ are positive cross-modal data points $o_P^t(o_P^v)$, and the remaining ones are negative cross-modal data points $o_N^t(o_N^v)$.

Following the work [37] to train the local model via a similar querying process, the well-trained adversarial generator $\mathcal{G}$ setting in the Cross-Gen will generate the effective adversarial cross-modal data point. Similarly, the Cross-Gen will inherit the decision knowledge of the target model and can easily destroy the relationship of the original hash space. Furthermore, we will optimize the parameter of δ_g by the adversarial generator $\mathcal{G}$ and initialize the discriminator $\mathcal{D}_{\delta_d}$ with the target model, which will be fixed while training.

As previously noted regarding the three-point goal of the model, to guarantee the attacking quality of adversarial examples generated by the adversarial generator $\mathcal{G}$, we set the objective function $\mathcal{L}_{gen}$ of adversarial generator $\mathcal{G}$, which can be defined as

$$\underset{\delta_g}{argmin}\, \mathcal{L}_{gen} = \sum_{\{o_A, o_P, o_N\} \in S_{tri}} (\mathcal{J}_{adv} + \beta \mathcal{J}_{qua} + \gamma \mathcal{J}_{rec}) \tag{4}$$

β and γ are the weighting constants to balance the term in Equation (4). S_{tri} is the triplet dataset that is constructed based on the target model. The main goal of Equation (4) is to obtain the optimal parameter δ_g^* of adversarial generator $\mathcal{G}$. $\mathcal{J}_{adv}$ is the adversarial loss function, $\mathcal{J}_{qua}$ is the quantization loss function, and $\mathcal{J}_{rec}$ is the reconstruction loss function. Next, we will introduce the above three loss functions below, respectively.

Adversarial loss function $\mathcal{J}_{adv}$ aims to minimize the Hamming distance between the hash code of query data point o_A and negative data point o_N, and maximize the Hamming distance between the hash code of query data point o_A and positive data point o_P. Hence, we can formulate the $\mathcal{J}_{adv}$ as follows:

$$\mathcal{J}_{adv} = \texttt{Ham}(\mathcal{D}(\mathcal{G}(o_A)), H_N) - \texttt{Ham}(\mathcal{D}(\mathcal{G}(o_A)), H_P) \tag{5}$$

where $H_P = \texttt{sign}(\mathcal{D}(o_P))$ and $H_N = \texttt{sign}(\mathcal{D}(o_N))$. $\{o_A, o_P, o_N\}$ are the anchor data point, positive data point, and negative data point, respectively.

Reconstruction loss function $\mathcal{J}_{re}$ minimizes the difference between the adversarial cross-modal data point and the original, e.g., to minimize the pixel difference between the adversarial image and the original image. We adopt the general L_2-norm loss to measure the reconstruction error as follows:

$$\mathcal{J}_{re} = ||o_A - \mathcal{G}(o_A)||_2^2 \tag{6}$$

where $\mathcal{G}(o_A)$ denote the adversarial cross modal data; $||\cdot||_2$ is the l_2 norm function. In fact, many norm functions are available, such as the l_∞ norm function and so on. Here, we construct a reconstruction loss function based on the l_2 norm function to ensure that the function is differentiable for updating the parameters of adversarial generator $\mathcal{G}_{\delta_g}$.

Quantization loss function $\mathcal{J}_{qua}$ reduces quantization errors between binary-like representations and binary hash codes, and we can formulate it as follows:

$$\mathcal{J}_{qua} = ||H_A - B_A||_2^2 \tag{7}$$

where $B_A = \texttt{sign}(H_A)$ and H_A is the binary-like representation of data point o_A; i.e., $H_A = \mathcal{D}(o_A)$. We create cross-modal adversarial examples to generate more compact hash codes in the target model to improve the attack effect by minimizing the quantization loss. We adopt the tanh-based quantization loss to approximate the sign function during backpropagation. No explicit orthogonality or balance constraints are imposed as the adversarial training itself encourages diverse and discriminative hash codes.

3.3. Implementation

We implement all experiments based on a 32-bit hash code and run on a server equipped with two GeForce RTX 2080Ti GPUs. We use the already constructed target cross-modal hashing model as the discriminator $\mathcal{D}$ and keep its parameters fixed throughout training. In cross-modal data retrieval, we consider two standard tasks: *image-to-text retrieval* (query: image; database: text) and *text-to-image retrieval* (query: text; database: image).

- For the **image-to-text task**, we build the adversarial image generator $\mathcal{G}_{\text{img}}$ using a modified AlexNet or ResNet-34/50. All backbones are initialized with ImageNet pre-trained weights, and the final classification layer is replaced by a deconvolutional head that outputs perturbed images of the same size as the input (e.g., $224 \times 224 \times 3$). The generator takes a clean image $\mathbf{x}$ and produces an adversarial version $\hat{\mathbf{x}} = \mathbf{x} + \boldsymbol{\delta}$, where the perturbation $\boldsymbol{\delta}$ is constrained by $\|\boldsymbol{\delta}\|_\infty \leq \epsilon = 8/255$ to ensure visual imperceptibility.
- For the **text-to-image task**, we construct the adversarial text generator $\mathcal{G}_{\text{text}}$ based on a 2-layer LSTM [38] with 512 hidden units per layer. The input is the original bag-of-words (BoW) vector, and the output is a perturbed textual representation in the same embedding space. Since text perturbations operate in the continuous feature domain, no explicit norm constraint is applied.

To train the Cross-Gen framework, we construct cross-modal triplets as follows: for each query, we retrieve 10 positive pairs (smallest Hamming distance in hash space) and 10 negative pairs (largest Hamming distance) from the database to form (q, p, n) triplets. The entire Cross-Gen network is optimized using the Adam optimizer [23] with $\beta_1 = 0.9$, $\beta_2 = 0.999$, and weight decay 10^{-4}, with an initial learning rate of 10^{-2} (decayed by a factor of 0.1 every 20 epochs). We train for 50 epochs and only update the parameters θ_g of the generator $\mathcal{G}$ while keeping the discriminator $\mathcal{D}$ frozen. The hyperparameter β in Equation (4) (balancing adversarial loss and perceptual loss) is set to 1×10^{-1}, chosen via grid search over $\{10^{-3}, 10^{-2}, 10^{-1}, 1\}$ on the validation set.

4. Experiments

In this section, we present the procedure used for the experiments of our proposed Cross-Gen. We exploit the widely used Mean Average Precision (MAP) and perceptibility (PER) evaluation metrics to measure the attack performance and perceptibility of the

generated adversarial examples. Finally, we show the time consumption of Cross-Gen in generating adversarial examples to demonstrate its superiority.

4.1. Datasets

In our experiment, we evaluate our method on two popular datasets, NUS-WIDE [25] and FLICKR-25K [24]. One dataset is NUS-WIDE, which consists of 190,421 image–text pairs in 81 categories, divided into retrieval, training, and query parts with the numbers 188,321, 10,500, and 2100. FLICKR-25K contains 20,015 image–text pairs, with 38 classes. Similarly, it is divided into three sections: 18,015, 10,000, and 2000. To construct the stolen dataset for adversarial attack generation, we randomly sample 1000 query pairs from each dataset. This choice is motivated by two practical considerations:

- Statistical significance: 1000 samples provide sufficient diversity to cover multiple semantic categories while ensuring stable evaluation of attack success rate.
- Computational feasibility: Generating high-quality adversarial examples is resource-intensive, and 1000 queries represent a commonly adopted scale in recent adversarial cross-modal studies [36].

The entire stolen set (i.e., all 1000 sampled pairs) is then used to craft adversarial perturbations targeting the victim hashing model.

4.2. Evaluation Metrics

To evaluate the performance of our framework, we follow the work [36] to set the Mean Average Precision (MAP). Further, the Average Precision (AP) can be calculated as

$$\texttt{AP} = \frac{1}{N} \times \sum_{i=1}^{N} \frac{i}{pos(i)}$$

where N is the size of the retrieval list. $pos(i)$ indicates the position of the i-th correct retrieval result in the returned list. Hence, the Mean Average Precision (`MAP`) is the mean value of the average accuracy (`AP`) of multiple retrieval results and is widely used in cross-modal Hamming retrieval. The lower the `MAP` value, the stronger the attack performance. Moreover, we follow the work [9] to measure the cross-modal data difference between adversarial and query data by calculating `PER` as

$$\texttt{PER} = \sqrt{\frac{1}{M}||o - o'||_2^2}$$

where $\mathbf{o}$ denotes the original cross-modal data and $\mathbf{o}'$ represents its adversarially perturbed version. Here, M is the total number of dimensions of the data (e.g., the total number of pixels for an image). A lower PER value indicates smaller distortion between the adversarial and original samples, implying better visual imperceptibility.

4.3. Compared Method

We deploy an advanced cross-modal hash retrieval method, CMHR [5], to train the target model cross-modal Hamming retrieval on both datasets. Specifically, we use AlexNet [39], ResNet-34, and ResNet-50 [18] as the backbone for image modality retrieval, and the text retrieval network is constructed through three fully connected layers. We set the training epoch as 200, and its data batch size is 32. To evaluate our performance, we follow the work [36] to set the AACH attack method for comparison, which is similar to ours. AACH demonstrates competitive performance relative to the advanced white-box attack methods [40]; hence, we use AACH as our comparison method. Specifically, we build an image retrieval module with vgg-f as the backbone and a text retrieval module

with three fully connected layers for the substitute model. AACH runs 500 iterations to obtain the adversarial attack based on the target model.

4.4. Results

Table 1 shows the MAP metrics for the two retrieval tasks, where "I→T" represents retrieving text through images and "T→I" represents retrieving images using text. Original represents the regular retrieval performance. We show the MAP values of AACH and our method on two datasets. The lower the MAP value, the better the attack performance. From Table 1, we can obtain the following conclusions: (1) For the AACH method, the attack performance will gradually improve as the number of iterations increases. For example, in the "I→T" task in FLICKR-25K, the MAP for 100 iterations is 59.97%, but the attack performance is only improved by 1.37% when it reaches 500 iterations. This shows the flaw of the instance-specific iterative attack method that it needs to consume significant computing resources, and the effect of the attack is not significantly improved. (2) Comparing two retrieval tasks on the FLICKR-25K and NUS-WIDE datasets, our method outperforms AACH. For example, in the ResNet-34 target model, our method in the $I \rightarrow T$ task is higher than the existing methods on the two datasets by 2.54% and 3.02%, respectively. Moreover, in both datasets, our method is higher by 0.75% and 3.11% in the "T→I" task. Finally, a desirable feature is that we can increase the speed of generating adversarial examples while improving the attack performance. We only need the trained generator for one forward pass to generate adversarial examples with a high attack rate. This is more efficient than running backpropagation multiple times on the target model.

Table 1. MAP on the attack methods with the 32-bit hash code on two image datasets. The best results are marked in bold.

Tasks	Method	Iteration	FLICKER-25k		
			AlexNet	ResNet-34	ResNet-50
I→T	AACH	Original	74.40%	80.24%	79.78%
		100	59.97%	64.15%	63.43%
		500	58.60%	63.02%	62.98%
	Ours	1	**56.21%**	**60.58%**	**60.24%**
T→I	AACH	Original	71.87%	77.58%	76.81%
		100	59.59%	67.47%	62.84%
		500	58.64%	66.68%	61.80%
	Ours	1	**57.83%**	**65.93%**	**60.50%**
Tasks	**Method**	**Iteration**	**NUS-WIDE**		
			AlexNet	**ResNet-34**	**ResNet-50**
I→T	AACH	Original	56.25%	66.04%	65.97%
		100	33.93%	40.16%	41.47%
		500	33.52%	39.94%	40.52%
	Ours	1	**30.13%**	**36.92%**	**39.43%**
T→I	AACH	Original	54.08%	56.95%	59.14%
		100	36.71%	47.07%	47.91%
		500	36.55%	46.37%	46.73%
	Ours	1	**34.33%**	**43.26%**	**45.58%**

4.5. Running Time and Imperceptibility

To better demonstrate the superiority of Cross-Gen in terms of generation speed and imperceptibility of adversarial examples, Table 2 shows the 32-bit hash code performance on two datasets, PER, as well as the running speed of AACH and Cross-Gen. We can observe that Cross-Gen has the fastest generation speed. We observe that the PER value of

AACH decreases gradually with the increase in the number of iterations, which shows that the iterative attack method can generate enough adversarial examples to be concealed. It is worth noting that Cross-Gen can also produce high-visual-quality and covert adversarial examples. We show examples in Figure 3 of two adversarial attack methods (i.e., iterative- and generative-based attack methods) and two categories of cross-modal retrieval tasks (i.e., image retrieval text and text retrieval image).

Table 2. Running time (min) and perceptibility (PER) on the attack methods with the 32-bit hash code on two image datasets. The best results are marked in bold.

Tasks	Method	Iteration	FLICKER-25k Time	FLICKER-25k Per	Method	Iteration	NUS-WIDE Time	NUS-WIDE Per
I→T	AACH	100	0.31	1.14	AACH	100	0.34	1.28
		500	1.27	0.93		500	1.44	1.19
	Ours	1	**0.011**	1.95	Ours	1	**0.006**	1.77
T→I	AACH	100	0.98	1.84	AACH	100	0.56	1.30
		500	3.81	1.67		500	2.03	1.26
	Ours	1	**0.004**	2.06	Ours	1	**0.009**	1.98

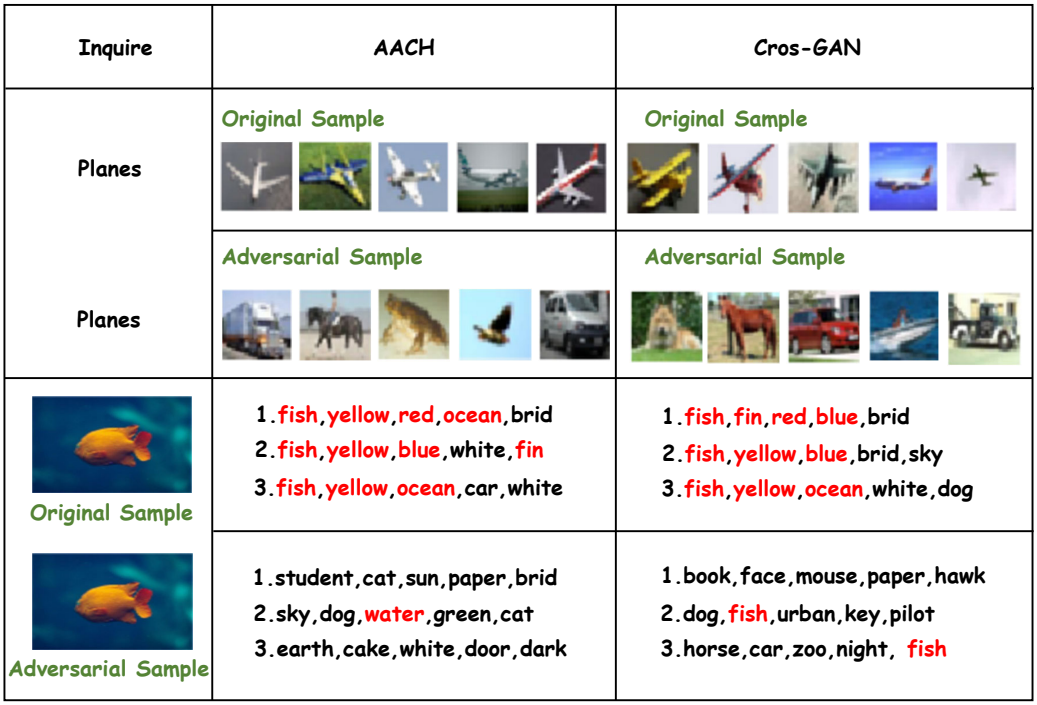

Figure 3. The illustration of cross-modal adversarial examples on two retrieval tasks (i.e., image retrieval text and text retrieval image).

5. Related Work

In this section, we introduce the main work of CMHR and describe the necessity of conducting a generative-based adversarial attack method.

5.1. Deep Cross-Modal Hamming Retrieval

Cross-modal search engines enable users to retrieve semantically relevant content by uploading either images or text queries. For instance, Google processed 89.3 billion searches in a single month in 2022 (https://www.oberlo.com/blog/google-search-statistics; accessed on 23 October 2025), while Baidu reported a user base of 795 million in 2021 (https://www.chyxx.com/industry/202109/973941.html; accessed on 23 October 2025). Given the scale of such multimodal data, traditional feature extraction methods—such as stroke

penetration, spectral analysis, and directional line elements—are inadequate for achieving efficient cross-modal retrieval. Consequently, recent research has shifted toward deep neural network-based approaches for cross-modal retrieval.

Studies have demonstrated the effectiveness of deep networks in mapping heterogeneous modalities into a shared hash space, significantly advancing retrieval performance [30,31,35]. Despite substantial progress in single-modality and basic cross-modal retrieval, challenges remain in designing robust cross-modal hash retrieval systems, particularly in aligning feature representations from disparate modalities into a unified semantic space.

To address this, Ref. [3] introduces a deep hybrid architecture that learns a joint visual–semantic embedding space for images and text. Ref. [4] generates compact binary hash codes to mitigate precision loss caused by relaxation during optimization, thereby improving retrieval accuracy. Ref. [5] proposes a novel deep hashing method that incorporates a pairwise focal loss based on an exponential distribution to refine the hashing objective. Ref. [41] constructs a joint matrix to integrate cross-modal semantic information and capture latent semantic similarities. More recently, Ref. [42] enhances cross-modal retrieval by modeling one-to-many embedding relationships through probabilistic cross-modal embeddings.

5.2. Adversarial Attacks on Deep Cross-Modal Hamming Retrieval

The successful application of deep neural networks (DNNs) has significantly enhanced the retrieval performance of cross-modal hash retrieval (CMHR) models. However, recent studies [6–8] have shown that well-crafted adversarial examples can effectively degrade DNN performance. CMHR systems, being DNN-based, inherit this vulnerability and are similarly susceptible to adversarial perturbations. Such attacks are typically categorized as white-box or black-box depending on whether the adversary has access to the target model's parameters.

In a white-box setting, Ref. [40] generates adversarial examples for CMHR by minimizing intra-modal similarity and maximizing inter-modal similarity. Ref. [43] proposes a decoupling strategy that separates cross-modal data into modality-dependent and modality-invariant components to craft adversarial inputs. The most closely related work to ours[36], constructs adversarial examples through iterative optimization using a stolen triplet dataset. As previously discussed, however, such iterative methods are computationally expensive.

In contrast, this paper investigates generative-based adversarial attacks in unsupervised settings. To the best of our knowledge, the existing generative approaches [20,21] have been limited to single-modality data and have rarely been explored in the context of cross-modal retrieval.

6. Conclusions

Adversarial sample generation for multimodal data can significantly facilitate research on high-dimensional visualization tasks. Prior work typically employed iterative adversarial attacks to generate examples against target models; however, such approaches are computationally inefficient and often require access to target labels, which is impractical in real-world settings. In contrast, this paper proposes an adversarial optimization framework based on generator networks. To the best of our knowledge, we are the first to construct a triplet database by querying the target model to obtain semantically aligned data pairs and subsequently train a parallel architecture comprising adversarial image and text generators. Our well-trained generator can efficiently produce adversarial examples with strong performance guarantees. Moreover, the method is straightforward to deploy and provides

a practical foundation for researchers to scale adversarial evaluation to large cross-modal retrieval datasets.

Meanwhile, the explorations in this study also offer useful insights for improving the use of multimodal data in comprehensive quality evaluation. The proposed methods help enhance the stability and reliability of multimodal data processing, thereby laying a foundation for further refinements in evaluation practices and the development of more robust analytical tools in related research contexts.

For future work, we aim to improve the accuracy of the acquired cross-modal data to better approximate decision boundaries and enhance the cross-domain transferability of the adversarial generator. Indeed, the current training process may be constrained by the use of randomly sampled cross-modal queries to collect semantic pairs from the target model—a limitation we identify as a key challenge for future investigation.

Author Contributions: Conceptualization, C.L., J.L. and R.S.; Methodology, C.L.; Validation, Y.Y. and R.S.; Formal analysis, S.L. and R.S.; Data curation, Y.Z.; Visualization, C.L.; Supervision, C.L. and R.S.; Project administration, C.H.; Funding acquisition, R.S. and L.C.; Writing—review and editing, C.L. and R.S. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported in part by "The 14th Five-Year Plan" Research Base for Education Science, Hunan, P.R. China Project (XJK23AJD021), Hunan Social Science Foundation (22YBA012), National Natural Science Foundation (62477046, 62177046), National Key R&D Program of China (2021YFC3340800), Chang-sha Science and Technology Key Project (kh2401027), Natural Science Foundation of Hunan Province (2023JJ40771), and the High Performance Computing Center of Central South University.

Data Availability Statement: The original contributions presented in this study are included in the article. Further inquiries can be directed to the corresponding author.

Conflicts of Interest: Author Yu Zhan was employed by the company China Telecom. The remaining authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

References

1. Hong, F.; Liu, C.; Yuan, X. DNN-VolVis: Interactive volume visualization supported by deep neural network. In Proceedings of the 2019 IEEE Pacific Visualization Symposium (PacificVis), Bangkok, Thailand, 23–26 April 2019; pp. 282–291.
2. Becker, M.; Lippel, J.; Stuhlsatz, A.; Zielke, T. Robust dimensionality reduction for data visualization with deep neural networks. *Graph. Model.* **2020**, *108*, 101060. [CrossRef]
3. Cao, Y.; Long, M.; Wang, J.; Yang, Q.; Yu, P.S. Deep visual-semantic hashing for cross-modal retrieval. In Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, San Francisco, CA, USA, 13–17 August 2016; pp. 1445–1454.
4. Jiang, Q.Y.; Li, W.J. Deep cross-modal hashing. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 3232–3240.
5. Cao, Y.; Liu, B.; Long, M.; Wang, J. Cross-modal hamming hashing. In Proceedings of the European Conference on Computer Vision (ECCV), Munich, Germany, 8–14 September 2018; pp. 202–218.
6. Shi, Y.; Wang, S.; Han, Y. Curls & whey: Boosting black-box adversarial attacks. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 15–20 June 2019; pp. 6519–6527.
7. Guo, C.; Gardner, J.; You, Y.; Wilson, A.G.; Weinberger, K. Simple black-box adversarial attacks. In Proceedings of the International Conference on Machine Learning, Long Beach, CA, USA, 9–15 June 2019; pp. 2484–2493.
8. Komkov, S.; Petiushko, A. Advhat: Real-world adversarial attack on arcface face id system. In Proceedings of the 2020 25th International Conference on Pattern Recognition (ICPR), Milan, Italy, 10–15 January 2021; pp. 819–826.
9. Szegedy, C.; Zaremba, W.; Sutskever, I.; Bruna, J.; Erhan, D.; Goodfellow, I.; Fergus, R. Intriguing properties of neural networks. *arXiv* **2013**, arXiv:1312.6199.
10. Moosavi-Dezfooli, S.M.; Fawzi, A.; Frossard, P. Deepfool: A simple and accurate method to fool deep neural networks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 2574–2582.

11. Dusmanu, M.; Schonberger, J.L.; Sinha, S.N.; Pollefeys, M. Privacy-preserving image features via adversarial affine subspace embeddings. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Nashville, TN, USA, 20–25 June 2021; pp. 14267–14277.
12. Che, Z.; Borji, A.; Zhai, G.; Ling, S.; Guo, G.; Callet, P.L. Adversarial attacks against deep saliency models. *arXiv* **2019**, arXiv:1904.01231. [CrossRef]
13. Ilyas, A.; Engstrom, L.; Athalye, A.; Lin, J. Black-box adversarial attacks with limited queries and information. In Proceedings of the International Conference on Machine Learning, Stockholm, Sweden, 10–15 July 2018; pp. 2137–2146.
14. Bhagoji, A.N.; He, W.; Li, B.; Song, D. Practical black-box attacks on deep neural networks using efficient query mechanisms. In Proceedings of the European Conference on Computer Vision (ECCV), Munich, Germany, 8–14 September 2018; pp. 154–169.
15. Chen, J.; Jordan, M.I.; Wainwright, M.J. Hopskipjumpattack: A query-efficient decision-based attack. In Proceedings of the 2020 IEEE Symposium on Security and Privacy (sp), San Francisco, CA, USA, 18–20 May 2020; pp. 1277–1294.
16. Li, H.; Xu, X.; Zhang, X.; Yang, S.; Li, B. Qeba: Query-efficient boundary-based blackbox attack. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 13–19 June 2020; pp. 1221–1230.
17. Na, Y.; Kim, J.H.; Lee, K.; Park, J.; Hwang, J.Y.; Choi, J.P. Domain adaptive transfer attack-based segmentation networks for building extraction from aerial images. *IEEE Trans. Geosci. Remote Sens.* **2020**, *59*, 5171–5182. [CrossRef]
18. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep residual learning for image recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 770–778.
19. Tang, Y.; Pino, J.; Li, X.; Wang, C.; Genzel, D. Improving speech translation by understanding and learning from the auxiliary text translation task. *arXiv* **2021**, arXiv:2107.05782. [CrossRef]
20. Li, S.; Neupane, A.; Paul, S.; Song, C.; Krishnamurthy, S.; Roy-Chowdhury, A.K.; Swami, A. Stealthy Adversarial Perturbations Against Real-Time Video Classification Systems. In Proceedings of the NDSS'19, San Diego, CA, USA, 24–27 February 2019.
21. Wang, X.; Zhang, Z.; Wu, B.; Shen, F.; Lu, G. Prototype-supervised adversarial network for targeted attack of deep hashing. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Nashville, TN, USA, 20–25 June 2021; pp. 16357–16366.
22. Hu, W.; Tan, Y. Generating adversarial malware examples for black-box attacks based on GAN. *arXiv* **2017**, arXiv:1702.05983. [CrossRef]
23. Kingma, D.P.; Ba, J. Adam: A method for stochastic optimization. *arXiv* **2014**, arXiv:1412.6980.
24. Huiskes, M.J.; Lew, M.S. The mir flickr retrieval evaluation. In Proceedings of the 1st ACM International Conference on Multimedia Information Retrieval, Vancouver, BC, Canada, 30–31 October 2008; pp. 39–43.
25. Chua, T.S.; Tang, J.; Hong, R.; Li, H.; Luo, Z.; Zheng, Y. Nus-wide: A real-world web image database from national university of singapore. In Proceedings of the ACM International Conference on Image and Video Retrieval, Santorini, Greece, 8–10 July 2009; pp. 1–9.
26. Weiss, Y.; Torralba, A.; Fergus, R. Spectral hashing. *Adv. Neural Inf. Process. Syst.* **2008**, *21*, 1753–1760.
27. Pushpalatha., K.R.; Chaitra., M.; Karegowda, A.G. Color Histogram based Image Retrieval—A Survey. *Int. J. Adv. Res. Comput. Sci.* **2013**, *4*, 119.
28. Zhang, Y.; Jin, R.; Zhou, Z.H. Understanding bag-of-words model: A statistical framework. *Int. J. Mach. Learn. Cybern.* **2010**, *1*, 43–52. [CrossRef]
29. Xu, R.; Li, C.; Yan, J.; Deng, C.; Liu, X. Graph Convolutional Network Hashing for Cross-Modal Retrieval. In Proceedings of the International Joint Conference on Artificial Intelligence (IJCAI), Macao, China, 10–16 August 2019; Volume 2019, pp. 982–988.
30. Wang, J.; Zhang, T.; Sebe, N.; Shen, H.T. A survey on learning to hash. *IEEE Trans. Pattern Anal. Mach. Intell.* **2017**, *40*, 769–790. [CrossRef] [PubMed]
31. Liu, X.; Mu, Y.; Zhang, D.; Lang, B.; Li, X. Large-scale unsupervised hashing with shared structure learning. *IEEE Trans. Cybern.* **2014**, *45*, 1811–1822. [CrossRef] [PubMed]
32. Qin, Q.; Huang, L.; Wei, Z.; Xie, K.; Zhang, W. Unsupervised deep multi-similarity hashing with semantic structure for image retrieval. *IEEE Trans. Circuits Syst. Video Technol.* **2020**, *31*, 2852–2865. [CrossRef]
33. Wang, X.; Shi, Y.; Kitani, K.M. Deep supervised hashing with triplet labels. In Proceedings of the Asian Conference on Computer Vision, Taipei, Taiwan, 20–24 November 2016; Springer: Berlin/Heidelberg, Germany, 2016; pp. 70–84.
34. Liu, H.; Wang, R.; Shan, S.; Chen, X. Deep supervised hashing for fast image retrieval. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 2064–2072.
35. Yan, C.; Xie, H.; Yang, D.; Yin, J.; Zhang, Y.; Dai, Q. Supervised hash coding with deep neural network for environment perception of intelligent vehicles. *IEEE Trans. Intell. Transp. Syst.* **2017**, *19*, 284–295. [CrossRef]
36. Li, C.; Gao, S.; Deng, C.; Liu, W.; Huang, H. Adversarial Attack on Deep Cross-Modal Hamming Retrieval. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Montreal, QC, Canada, 10–17 October 2021; pp. 2218–2227.

37. Papernot, N.; McDaniel, P.; Goodfellow, I.; Jha, S.; Celik, Z.B.; Swami, A. Practical black-box attacks against machine learning. In Proceedings of the 2017 ACM on Asia Conference on Computer and Communications Security, Abu Dhabi, United Arab Emirates, 2–6 April 2017; pp. 506–519.
38. Hochreiter, S.; Schmidhuber, J. Long short-term memory. *Neural Comput.* **1997**, *9*, 1735–1780. [CrossRef] [PubMed]
39. Krizhevsky, A.; Sutskever, I.; Hinton, G.E. Imagenet classification with deep convolutional neural networks. *Adv. Neural Inf. Process. Syst.* **2012**, *25*, 1097–1105. [CrossRef]
40. Li, C.; Gao, S.; Deng, C.; Xie, D.; Liu, W. Cross-modal learning with adversarial samples. *Adv. Neural Inf. Process. Syst.* **2019**, *32*, 10792–10802.
41. Su, S.; Zhong, Z.; Zhang, C. Deep joint-semantics reconstructing hashing for large-scale unsupervised cross-modal retrieval. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Seoul, Korea, 27 October–2 November 2019; pp. 3027–3035.
42. Chun, S.; Oh, S.J.; De Rezende, R.S.; Kalantidis, Y.; Larlus, D. Probabilistic embeddings for cross-modal retrieval. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Nashville, TN, USA, 20–25 June 2021; pp. 8415–8424.
43. Li, C.; Tang, H.; Deng, C.; Zhan, L.; Liu, W. Vulnerability vs. reliability: Disentangled adversarial examples for cross-modal learning. In Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, Virtual Event, USA, 23–27 August 2020; pp. 421–429.