MAKE EVERY BIT COUNT: RESIDUAL CAPACITY REALLOCATION FOR UNSUPERVISED CROSS-MODAL HASHING

Anonymous authors
Paper under double-blind review

ABSTRACT

Unsupervised cross-modal hashing (UCMH) maps heterogeneous modalities into compact Hamming spaces for efficient retrieval. However, existing methods typically optimize a fixed-length hash code as a whole, overlooking that individual bits may differ substantially in marginal utility while some cross-modal relations remain insufficiently represented. This leads to capacity misallocation under a fixed hash budget. To address this issue, we propose ReCapHash, a plug-in framework for fixed-budget residual capacity reallocation. ReCapHash first estimates the bidirectional marginal utility of individual hash dimensions and adaptively identifies reclaimable low-utility bits while preserving the remaining dimensions as an informative core. It then contrasts pretrained semantic affinity with the Hamming affinity captured by the retained core to mine unresolved cross-modal relations and organize them into a paired residual graph. Global spectral partitioning and diversity-aware binary recovery transform these relations into complementary residual targets, which are distilled into modality-specific predictors and reallocated to the reclaimed positions. Extensive experiments on MIRFlickr, NUS-WIDE, and IAPR-TC12 demonstrate strong retrieval performance across multiple code lengths and retrieval directions, while consistent improvements over diverse base hashing methods further validate the generality of ReCapHash.

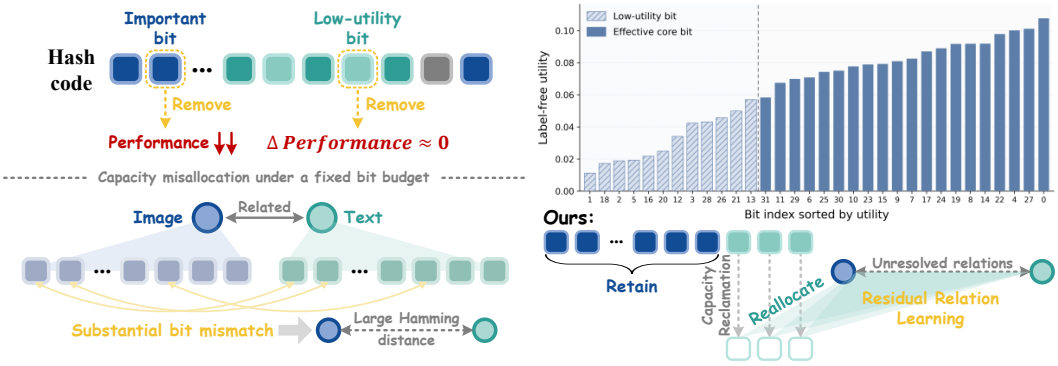


Figure 1: Motivation for fixed-budget residual capacity reallocation. Hash dimensions exhibit unequal marginal utility, as illustrated by the 32-bit utility profile on MIRFlickr (upper right), while related image–text pairs may remain poorly aligned in Hamming space. These observations motivate reclaiming low-utility coordinates and reallocating them to unresolved cross-modal relations.

1 INTRODUCTION

Cross-modal retrieval aims to identify semantically related instances across heterogeneous modalities, such as images and texts, and has become important for large-scale multimodal search. Recent studies further consider retrieval reliability through methods such as FUME (Duan et al., 2025)

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

and neighborhood bias in representation learning (Lin et al., 2025). With the rapid growth of vision–language data, directly storing and comparing high-dimensional real-valued representations becomes increasingly costly. Cross-modal hashing addresses this issue by mapping heterogeneous representations into a compact Hamming space (Tan et al., 2026), where binary codes enable efficient storage and fast bit-wise similarity computation. Recent studies such as SCH (Hu et al., 2024) further show that the organization of Hamming space itself critically affects retrieval, while distribution-aware hashing (Sun et al., 2024) and diverse-instance matching (Tu et al., 2025) improve the semantic organization of binary representations.

Cross-modal hashing generally includes supervised and unsupervised paradigms. Supervised methods learn discriminative codes from semantic annotations (Liu et al., 2024), with recent work considering noisy supervision through robust self-paced learning (Pu et al., 2025). Since reliable annotations are costly at scale, unsupervised cross-modal hashing (UCMH) has received increasing attention. Recent UCMH methods exploit multimodal knowledge graphs (Liang et al., 2024), multi-granularity Transformer alignment (Wang et al., 2024), vision-guided community structures (Fan & Cao, 2025), similarity-fusion Transformers (Yang et al., 2025), and formal-concept associations (Li et al., 2025b). HINT (Xiao et al., 2026) models hierarchical relations, ENHash (Fu et al., 2026b) mines error-driven fine-grained knowledge, SCTH (Yang et al., 2026c) combines Transformer fusion with clustering, and BRAINHash (Fu et al., 2026a) introduces region-aligned interactions. Related studies also investigate trustworthy hashing (Li et al., 2025c), online retrieval (Kang et al., 2025; Li et al., 2025a), incomplete modalities (Chen et al., 2026; Zhang et al., 2026b), and long-tailed retrieval (Cao et al., 2026), reflecting increasingly diverse cross-modal hashing scenarios.

Despite this progress, most existing methods still optimize the available hash dimensions jointly as a single representation (Zou et al., 2025; Su et al., 2026b), leaving less attention to how an already learned fixed hash budget is internally utilized. Unequal informativeness across dimensions has motivated hash-bit selection (Liu et al., 2013), while adaptive bit selection (Wang et al., 2025) and variable-length hashing (Zuo et al., 2026) further adjust which dimensions are retained or how many bits are used. Recent work also revisits learned codes from other perspectives: regenerated cross-modal hashing (Jiang et al., 2026) transforms existing codes to arbitrary target lengths, while capacity-aware selective hashing (Liu et al., 2026) allocates limited bits in continual retrieval. In contrast, we consider a static fixed-budget setting and ask: *after a base B-bit representation has already been learned, can low-utility coordinates be reused to encode semantic relations that the effective code still fails to represent?*

The first challenge is *where capacity can be safely reclaimed*. As illustrated in Figure 1, removing some dimensions causes substantial retrieval degradation, whereas removing others has little effect, revealing large differences in marginal utility (Luo et al., 2026). The upper-right panel shows the actual utility profile of 32-bit base codes on MIRFlickr, indicating uneven capacity utilization. Rather than discarding low-utility bits, we regard their coordinates as reusable capacity. The second challenge is *what information should occupy the reclaimed positions*. Even when all B positions are occupied, some semantically related image–text pairs (Zhang et al., 2026c) remain poorly aligned in Hamming space. Existing methods improve hierarchical relations (Zhang et al., 2026a), error-driven knowledge mining (Fu et al., 2026b), clustering-based semantics (Zhao et al., 2026), generative similarity learning (Sun et al., 2026), and reliable alignment (Yang et al., 2026b). However, relearning semantic relations (Liu et al., 2025) without considering what the retained code already represents may repeatedly encode redundant structure. The key problem is therefore to reclaim low marginal-utility coordinates and reassign them specifically to relations left unexplained by the preserved code.

To address this problem, we propose ReCapHash, a plug-in framework for fixed-budget residual capacity reallocation in unsupervised cross-modal hashing. ReCapHash first measures bidirectional marginal utility to identify reclaimable coordinates and preserves the remaining dimensions as an effective core. It then contrasts pretrained semantic affinity with the Hamming affinity captured by the core to discover semantically strong yet insufficiently represented residual relations. These relations form a paired residual graph, from which global spectral recovery and diversity-aware binarization produce complementary residual targets. The targets are distilled into modality-specific predictors and reallocated to the reclaimed positions. ReCapHash thus forms a diagnose–discover–recover–reallocate process while keeping the final representation exactly B bits and leaving the base hash learner unchanged. Extensive experiments on MIRFlickr, NUS-WIDE, and IAPR-TC12 demonstrate strong retrieval performance across code lengths and retrieval directions, while consistent improvements over diverse base hashing methods validate its plug-in generality.

The main contributions of this paper are: ❶ **Fixed-Budget Residual Capacity Reallocation.** We study fixed-budget residual capacity reallocation by reclaiming low marginal-utility coordinates and reassigning them to semantic relations unexplained by the retained core. Unlike bit selection or variable-length hashing, reclaimed dimensions are reused rather than discarded, while the original code length remains unchanged. ❷ **Core-Conditioned Residual Recovery.** We develop a diagnose–discover–recover–reallocate framework in which marginal utility determines the reclaimable capacity, the retained core defines residual cross-modal relations, and complementary residual bits are recovered and written back to the reclaimed coordinates. ❸ **Plug-in Generality under a Fixed Budget.** ReCapHash improves diverse base UCMH representations without modifying their original learning objectives or increasing the hash budget, with consistent gains across datasets, code lengths, retrieval directions, and base hashing methods.

2 Related Works

Cross-modal Retrieval matches semantically related instances across heterogeneous modalities in a shared space. Recent work studies out-of-distribution generalization (Kalyan et al., 2024), multimodal query–document modeling (Caffagni et al., 2025), dense knowledge alignment (Cui et al., 2025), fine-grained image–text alignment (Asokan et al., 2025), and backward-compatible representations across evolving vision–language models (Jang & Lim, 2025). Cross-modal hashing further maps representations into compact binary codes for efficient Hamming-space search, where code-space organization strongly affects retrieval quality (Yang et al., 2026a).

Unsupervised Cross-modal Hashing learns binary representations without semantic annotations by exploiting intrinsic multimodal correlations. Existing methods model fine-grained relations through multimodal knowledge graphs (Liang et al., 2024), multi-granularity Transformers (Wang et al., 2024), community similarity (Fan & Cao, 2025), and similarity fusion (Kuai et al., 2025); more recent approaches explore hierarchical structures (Xiao et al., 2026), error-driven knowledge mining (Fu et al., 2026b), and reliable alignment (Yang et al., 2026b). Beyond standard UCMH, recent hashing studies also consider partial-label ambiguity (Su et al., 2026a), noisy multi-label supervision (Peng et al., 2026), adversarial robustness (Yu et al., 2026), expanding online label spaces (Fan et al., 2026), and polysemic semantic instances (Han et al., 2026). These developments mainly improve semantic relation learning, robustness, or modality alignment; ReCapHash instead examines how effectively a learned fixed-length code uses its limited capacity.

Cross-modal Relationship and Capacity Modeling is central to discriminative hashing (Tan et al., 2025). Graph-, community-, and hierarchy-based methods (Qin et al., 2025) organize cross-modal relations, while VH-ABL (Zuo et al., 2026) adapts the effective code length. RCMH (Jiang et al., 2026) regenerates learned codes to arbitrary lengths, and CASH (Liu et al., 2026) allocates limited bits for continual retrieval. In contrast, ReCapHash keeps the original B-bit budget fixed, reclaims low-utility coordinates, and conditions residual relation discovery on the retained core so that reclaimed capacity encodes complementary relations that remain unresolved by the preserved code.

3 Preliminary

Given N paired image-text samples $\mathcal{D} = \{(x_i^v, x_i^t)\}_{i=1}^N$ without semantic labels, pretrained cross-modal encoders provide aligned semantic representations $\mathbf{f}_i^v, \mathbf{f}_i^t \in \mathbb{R}^d$. An arbitrary base cross-modal hashing method employs modality-specific mappings $\phi_v(\cdot)$ and $\phi_t(\cdot)$ to project the two modalities into a shared B-bit Hamming space:

$$\mathbf{b}_i^v = \operatorname{sign}(\phi_v(\mathbf{f}_i^v)), \qquad \mathbf{b}_i^t = \operatorname{sign}\big(\phi_t(\mathbf{f}_i^t)\big), \tag{1}$$

where $\mathbf{b}_i^* \in \{-1, +1\}^B$, $* \in \{v, t\}$, and $\operatorname{sign}(\cdot)$ denotes the element-wise sign function. We collect all image and text codes as $\mathbf{B}^v, \mathbf{B}^t \in \{-1, +1\}^{N \times B}$.

For an image code $\mathbf{b}_i^v$ and a text code $\mathbf{b}_j^t$, their Hamming distance is

$$d_H(\mathbf{b}_i^v, \mathbf{b}_j^t) = \frac{1}{2}\left(B - (\mathbf{b}_i^v)^\top \mathbf{b}_j^t\right). \tag{2}$$

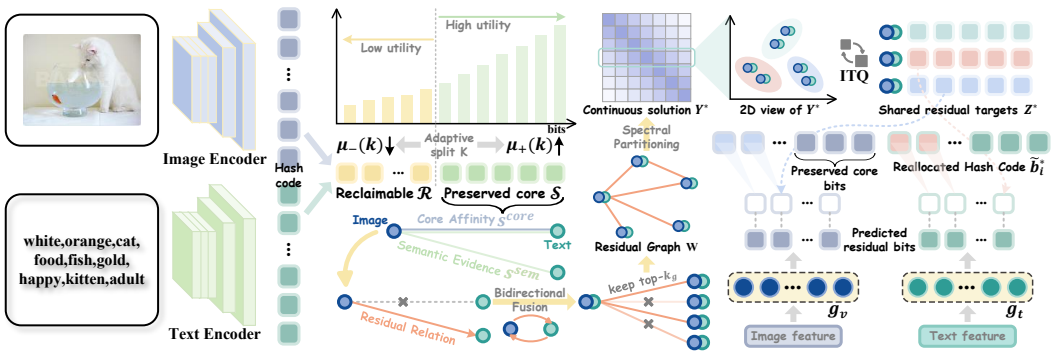


Figure 2: Overview of ReCapHash, which reclaims low-utility hash dimensions and constructs a core-conditioned residual graph to recover shared residual targets. It then distills them into modality-specific residual bits and reallocates them to reclaimed positions under the original hash budget.

4 METHODOLOGY

ReCapHash is a plug-in framework that reallocates under-utilized dimensions of an existing B-bit cross-modal hash code to semantic relations insufficiently represented by its preserved core. As illustrated in Figure 2, it identifies low-utility dimensions, constructs a core-conditioned residual graph, recovers complementary residual targets, and distills them into modality-specific replacement bits while preserving the original hash budget.

4.1 UTILITY-GUIDED ADAPTIVE CAPACITY RECLAMATION

We first ask *where* capacity can be reclaimed. Since reclaimable capacity varies across representations, we estimate the marginal utility of individual dimensions directly from the learned codes.

Bidirectional Marginal Utility. Let $\mathcal{B} = \{1, \ldots, B\}$ denote the complete set of hash dimensions. Using the pretrained semantic representations, we construct label-free cross-modal neighborhoods that provide semantic ranking references. For an image query i, cross-modal text candidates are ranked according to $s_{ij}^{\text{sem}} = \cos(\mathbf{f}_i^v, \mathbf{f}_j^t)$. We construct the semantic positive set $\mathcal{P}_i^t$ by including the paired text x_i^t and the highest-ranked remaining candidates until $|\mathcal{P}_i^t| = k_p$, and define $\mathcal{N}_i^t$ as the subsequent k_n highest-ranked candidates after excluding $\mathcal{P}_i^t$. The corresponding image neighborhoods for text queries are constructed analogously.

Given a subset of hash dimensions $\mathcal{S} \subseteq \mathcal{B}$, we define the image-to-text semantic–Hamming ranking agreement as

$$\mathcal{A}^{v\to t}(\mathcal{S}) = \frac{1}{N}\sum_{i=1}^{N}\frac{1}{|\mathcal{P}_i^t||\mathcal{N}_i^t|}\sum_{p\in\mathcal{P}_i^t}\sum_{n\in\mathcal{N}_i^t}\mathbb{I}\left[d_H^{\mathcal{S}}\left(\mathbf{b}_i^v, \mathbf{b}_p^t\right) < d_H^{\mathcal{S}}\left(\mathbf{b}_i^v, \mathbf{b}_n^t\right)\right], \tag{3}$$

where $d_H^{\mathcal{S}}(\cdot,\cdot)$ denotes Hamming distance computed only over dimensions indexed by $\mathcal{S}$ and $\mathbb{I}[\cdot]$ is the indicator function. The text-to-image agreement $\mathcal{A}^{t\to v}(\mathcal{S})$ is defined analogously by exchanging the query and candidate modalities.

We define the marginal utility of the b-th dimension as the agreement drop after its removal:

$$u_b^{v\to t} = \mathcal{A}^{v\to t}(\mathcal{B}) - \mathcal{A}^{v\to t}(\mathcal{B}\backslash\{b\}),\ u_b^{t\to v} = \mathcal{A}^{t\to v}(\mathcal{B}) - \mathcal{A}^{t\to v}(\mathcal{B}\backslash\{b\}),\ u_b = \frac{u_b^{v\to t} + u_b^{t\to v}}{2}. \tag{4}$$

A small u_b indicates that the dimension contributes little unique ranking information and is therefore potentially reclaimable.

Adaptive Capacity Reclamation. We order the dimensions according to their combined utilities. Let $\pi_1, \ldots, \pi_B$ denote the resulting permutation such that $u_{\pi_1} \le u_{\pi_2} \le \cdots \le u_{\pi_B}$. Instead of

prescribing a fixed replacement ratio, we locate the split that most clearly separates the low-utility portion from the informative portion of the ordered utility profile.

For a candidate split position k, let $\mu_-(k) = \frac{1}{k}\sum_{i=1}^{k} u_{\pi_i}$ and $\mu_+(k) = \frac{1}{B-k}\sum_{i=k+1}^{B} u_{\pi_i}$ denote the mean utilities of the two groups. We determine the reclaimable capacity by maximizing their weighted between-group separation:

$$K = \arg\max_{1\le k<B} \frac{k(B-k)}{B^2}\left[\mu_-(k) - \mu_+(k)\right]^2. \tag{5}$$

This yields a data-dependent reclaimable capacity K without a predefined replacement ratio. We define $\mathcal{R} = \{\pi_1, \ldots, \pi_K\}$ and $\mathcal{C} = \mathcal{B} \setminus \mathcal{R}$ as the reclaimable and preserved sets, respectively, yielding K residual dimensions and a $(B-K)$-bit core.

4.2 Core-Conditioned Residual Relation Graph Construction

Capacity reclamation determines *where* new information can be stored; we next ask *what* should occupy it. To avoid redundancy with the preserved core, we condition relation discovery on the core and focus on semantic relations that remain unexplained.

Core-Conditioned Residual Relation. For image i and text j, the pretrained semantic affinity s_{ij}^{sem} serves as the semantic reference, while the relation represented by the retained core is measured through its normalized Hamming affinity:

$$s_{ij}^{\mathrm{core}} = \frac{1}{B-K}\left(\mathbf{b}_{i,\mathcal{C}}^{v}\right)^{\top}\mathbf{b}_{j,\mathcal{C}}^{t}. \tag{6}$$

Because semantic and Hamming affinities may have different query-dependent scales, we standardize each affinity row before computing their discrepancy. For $q \in \{\mathrm{sem}, \mathrm{core}\}$, let $\bar{s}_{ij}^{q} = (s_{ij}^{q} - \mu_i^q)/(\sigma_i^q + \epsilon)$, where μ_i^q and σ_i^q are the mean and standard deviation of the corresponding affinity scores for query i. The image-to-text residual relation is then defined as

$$r_{ij}^{v\to t} = \left[\bar{s}_{ij}^{\mathrm{sem}} - \alpha\bar{s}_{ij}^{\mathrm{core}}\right]_+, \tag{7}$$

where $[x]_+ = \max(x, 0)$ and α controls how strongly information already represented by the core is suppressed. The text-to-image residual is defined analogously by exchanging the query and candidate modalities. We restrict each query to its top-k_s semantic candidates and retain large residuals as semantically supported yet core-unresolved relations.

Directional Residual Mining. For each image query, we retain the k_r largest residuals within its top-k_s semantic candidates to form $\mathcal{N}_i^{v\to t}$; $\mathcal{N}_i^{t\to v}$ is defined analogously. Here k_r controls the residual neighborhood size and k_s the semantic candidate pool.

Paired Residual Graph.

To obtain globally consistent targets from overlapping local neighborhoods, we represent each paired instance (x_i^v, x_i^t) as a node v_i in a residual graph. The directional residual evidence between paired nodes is aggregated as

$$A_{ij} = r_{ij}^{v\to t}\mathbb{I}\left[j \in \mathcal{N}_i^{v\to t}\right] + r_{ij}^{t\to v}\mathbb{I}\left[j \in \mathcal{N}_i^{t\to v}\right]. \tag{8}$$

Relations receiving evidence in both node directions provide stronger support that the corresponding unresolved structure is reliable. We therefore define $\mathbb{I}_{ij}^{\mathrm{rec}} = \mathbb{I}[A_{ij} > 0]\mathbb{I}[A_{ji} > 0]$ and enhance such reciprocal relations through

$$\widetilde{A}_{ij} = A_{ij}\left(1 + \beta\mathbb{I}_{ij}^{\mathrm{rec}}\right), \tag{9}$$

where β controls the strength of reciprocal enhancement.

The undirected residual affinity is first obtained by symmetrization:

$$W_{ij} = \frac{1}{2}\left(\widetilde{A}_{ij} + \widetilde{A}_{ji}\right). \tag{10}$$

To suppress weak long-tail connections, we retain the k_g strongest affinities in each row and then re-symmetrize the resulting sparse graph by $\mathbf{W} \leftarrow \max(\mathbf{W}, \mathbf{W}^{\top})$ element-wise. Here k_r controls the breadth of locally mined residual evidence, whereas k_g controls the sparsity of the global residual graph. Unlike a conventional semantic graph, the resulting $\mathbf{W}$ represents only semantic structure that remains insufficiently encoded by the effective core rather than all available semantic relations.

4.3 Global Residual-Bit Recovery

Given the residual graph, we recover exactly K globally coherent residual dimensions corresponding to the reclaimed capacity. Rather than treating local residual relations independently, we seek a global representation that consistently preserves their relational structure.

Global Spectral Recovery. Let $\mathbf{D}$ denote the degree matrix of the residual graph, with $D_{ii} = \sum_j W_{ij}$, and let $\mathbf{L} = \mathbf{D} - \mathbf{W}$ be its graph Laplacian. We seek K continuous partition directions that preserve strong residual connections while avoiding degenerate solutions:

$$\mathbf{Y}^* = \arg\min_{\mathbf{Y}\in\mathbb{R}^{N\times K}} \operatorname{Tr}(\mathbf{Y}^\top \mathbf{L}\mathbf{Y}),\ \text{s.t. } \mathbf{Y}^\top\mathbf{D}\mathbf{Y} = \mathbf{I},\ \mathbf{Y}^\top\mathbf{D}\mathbf{1} = \mathbf{0}. \tag{11}$$

The orthogonality constraint prevents collapse and separates different partition directions, while the second constraint removes the trivial constant solution. The resulting non-trivial spectral directions transform overlapping local semantic–Hamming discrepancies into a globally coherent residual representation.

Diversity-Aware Binary Recovery. Direct thresholding may incur substantial quantization error, so we employ an iterative quantization (ITQ)-style orthogonal rotation:

$$\min_{\mathbf{Z},\mathbf{R}} \|\mathbf{Z} - \mathbf{Y}^*\mathbf{R}\|_F^2,\ \text{s.t. } \mathbf{Z}\in\{-1,+1\}^{N\times K},\ \mathbf{R}^\top\mathbf{R} = \mathbf{I}. \tag{12}$$

We alternately update the binary codes by thresholding the rotated embedding and update $\mathbf{R}$ through orthogonal Procrustes optimization.

Different initial rotations can lead to different binary solutions with similar graph fidelity. Preserving the residual graph alone, however, does not guarantee that different recovered dimensions encode distinct residual information. For a candidate residual code matrix $\mathbf{Z}$, we measure inter-bit redundancy by the maximum absolute correlation between different residual dimensions:

$$\rho_{\text{bit}}(\mathbf{Z}) = \max_{p\neq q} \left|\operatorname{Corr}\left(\mathbf{z}_{:,p}, \mathbf{z}_{:,q}\right)\right|, \tag{13}$$

with $\rho_{\text{bit}}(\mathbf{Z}) = 0$ when $K = 1$.

A desirable residual representation should simultaneously preserve strong unresolved relations in the graph and avoid redundant recovered dimensions. We therefore define its residual-graph agreement as the weighted normalized Hamming similarity between connected graph nodes:

$$\mathcal{G}(\mathbf{Z};\mathbf{W}) = \frac{\displaystyle\sum_{i<j} W_{ij}\left(1 - \frac{d_H(\mathbf{z}_i,\mathbf{z}_j)}{K}\right)}{\displaystyle\sum_{i<j} W_{ij}}. \tag{14}$$

Given a set $\mathcal{Q}$ of candidate binary codes obtained from multiple orthogonal restarts, we select

$$\mathbf{Z}^* = \arg\max_{\mathbf{Z}\in\mathcal{Q}} \left[\mathcal{G}(\mathbf{Z};\mathbf{W}) - \lambda_d \rho_{\text{bit}}(\mathbf{Z})\right], \tag{15}$$

where λ_d controls the trade-off between preserving unresolved graph structure and reducing redundancy among recovered dimensions. The selected $\mathbf{Z}^* \in \{-1,+1\}^{N\times K}$ thus provides globally coherent and mutually complementary residual targets.

4.4 Cross-Modal Residual Distillation and Reallocation

The graph recovery stage produces residual targets only for paired training nodes. Cross-modal retrieval, however, requires unseen images and texts to generate their residual codes independently. We therefore distill the recovered graph structure into modality-specific predictors.

Cross-Modal Residual Distillation. Let $g_v(\cdot)$ and $g_t(\cdot)$ denote lightweight residual heads whose outputs are K-dimensional logits for the visual and textual modalities. Because graph node i represents the paired instance (x_i^v, x_i^t), both modalities share the same recovered residual target $\mathbf{z}_i^*$. We convert it to $\mathbf{y}_i^* = (\mathbf{z}_i^* + 1)/2$ and optimize

$$\mathcal{L}_{\text{dist}} = \frac{1}{2N}\sum_{i=1}^{N}\left[\text{BCEWithLogits}\left(g_v(\mathbf{f}_i^v), \mathbf{y}_i^*\right) + \text{BCEWithLogits}\left(g_t(\mathbf{f}_i^t), \mathbf{y}_i^*\right)\right]. \tag{16}$$

Table 1: Performance comparison of ReCapHash and baselines on the three datasets with various code lengths. Best results are shown in **bold** and second-best results are underlined.

Task	Method	Source	MIRFlickr				NUS-WIDE				IAPR-TC12			
			16 bits	32 bits	64 bits	128 bits	16 bits	32 bits	64 bits	128 bits	16 bits	32 bits	64 bits	128 bits
I→T	UCCH	TPAMI'22	0.8077	0.8149	0.8136	0.8374	0.7342	0.7429	0.7431	0.7579	0.5379	0.5837	0.5940	0.6202
	CIRH	TKDE'22	0.6293	0.6435	0.6396	0.6837	0.5165	0.6882	0.6700	0.6909	0.4278	0.5101	0.5270	0.5437
	UCH-STM	TMM'23	0.6556	0.7270	0.7618	0.7537	0.5419	0.5992	0.6951	0.7279	0.4467	0.5157	0.5385	0.5670
	HEH	MM'23	0.7733	0.8516	0.8563	0.8742	0.6814	0.7361	0.7589	0.7547	0.5687	0.5873	0.6185	0.6292
	CMGCH	AAAI'24	0.7879	0.8017	0.8406	0.8605	0.6555	0.7015	0.7480	0.7713	0.6070	0.6633	0.7155	0.7499
	GASKN	ICDE'24	0.8072	0.8369	0.8630	0.8773	0.7039	0.7415	0.7789	0.7980	0.6608	0.7228	0.7577	0.7743
	HuggingHash	IJCV'24	0.8288	0.8638	0.8891	0.9010	0.7217	0.7663	0.7938	0.8126	0.6559	0.7269	0.7567	0.7732
	VTM-UCH	AAAI'25	0.7773	0.7649	0.7937	0.8160	0.6802	0.7531	0.7859	0.8003	0.5789	0.6330	0.6567	0.6919
	USFTH	MM'25	0.7435	0.7667	0.7722	0.7962	0.6285	0.6744	0.6803	0.7104	0.5109	0.5683	0.5900	0.6107
	ENHash	AAAI'26	0.7886	0.8665	0.8842	0.8853	0.7390	0.7621	0.7702	0.7811	0.6691	0.6963	0.7114	0.7317
	BRAINHash	TIP'26	0.8498	0.8740	0.8947	0.8938	0.7696	0.7801	0.7954	0.8022	0.6362	0.7076	0.7234	0.7396
	UWMCH	CVPR'26	0.7611	0.7691	0.8082	0.8396	0.6200	0.6632	0.7079	0.7598	0.5810	0.6257	0.6830	0.7407
	HINT	ICLR'26	0.8153	0.8046	0.8109	0.8003	0.7164	0.7191	0.7122	0.7055	0.4896	0.5319	0.5105	0.5459
	ReCapHash	**Ours**	**0.8871**	**0.8977**	**0.9044**	**0.9051**	**0.7719**	**0.7953**	**0.8152**	**0.8131**	**0.6764**	**0.7631**	**0.7863**	**0.8042**
T→I	UCCH	TPAMI'22	0.7803	0.7995	0.8168	0.8206	0.7091	0.7202	0.7310	0.7289	0.5415	0.5842	0.6112	0.6152
	CIRH	TKDE'22	0.6131	0.6234	0.6429	0.6412	0.5226	0.6608	0.6442	0.6788	0.4235	0.5058	0.5311	0.5348
	UCH-STM	TMM'23	0.6452	0.7022	0.7409	0.7676	0.5425	0.5880	0.6664	0.7036	0.4537	0.5249	0.5522	0.5733
	HEH	MM'23	0.7565	0.7822	0.8101	0.8159	0.6596	0.6815	0.6937	0.6950	0.5420	0.5653	0.5923	0.5855
	CMGCH	AAAI'24	0.7626	0.7861	0.8190	0.8360	0.6512	0.6905	0.7273	0.7525	0.6103	0.6689	0.7155	0.7465
	GASKN	ICDE'24	0.7882	0.8169	0.8400	0.8528	0.6966	0.7249	0.7526	0.7616	0.6488	0.7140	0.7519	0.7658
	HuggingHash	IJCV'24	0.8140	0.8395	0.8551	0.8652	0.7080	0.7444	0.7703	0.7841	0.6487	0.7178	0.7430	0.7625
	VTM-UCH	AAAI'25	0.7677	0.7728	0.7936	0.7992	0.6701	0.7311	0.7532	0.7702	0.5628	0.6222	0.6510	0.6822
	USFTH	MM'25	0.6905	0.7255	0.7463	0.7633	0.5406	0.6052	0.6618	0.6709	0.5061	0.5519	0.5791	0.6084
	ENHash	AAAI'26	0.7429	0.8424	0.8429	0.8547	0.7183	0.7342	0.7491	0.7640	0.6513	0.6769	0.7008	0.7238
	BRAINHash	TIP'26	0.8277	0.8527	0.8605	0.8686	0.7195	0.7554	0.7666	0.7782	0.6344	0.6979	0.7177	0.7337
	UWMCH	CVPR'26	0.7499	0.7597	0.7899	0.8171	0.6162	0.6520	0.6981	0.7257	0.5858	0.6255	0.6792	0.7285
	HINT	ICLR'26	0.7924	0.7950	0.8019	0.7884	0.6893	0.6792	0.6925	0.7008	0.4736	0.5343	0.5193	0.5427
	ReCapHash	**Ours**	**0.8510**	**0.8626**	**0.8634**	**0.8701**	**0.7404**	**0.7632**	**0.7813**	**0.7881**	**0.6674**	**0.7497**	**0.7772**	**0.7902**

Table 2: Plug-in evaluation of ReCapHash across different base hashing methods. Parentheses denote absolute gains over the corresponding base performance in Table 1.

Task	Method	MIRFlickr				NUS-WIDE				IAPR-TC12			
		16 bits	32 bits	64 bits	128 bits	16 bits	32 bits	64 bits	128 bits	16 bits	32 bits	64 bits	128 bits
I→T	HEH	**0.8685** (+9.52)	**0.8952** (+4.36)	**0.9115** (+5.52)	**0.9133** (+3.91)	**0.7447** (+6.33)	**0.7913** (+5.52)	**0.8041** (+4.52)	**0.8110** (+5.63)	**0.6330** (+6.43)	**0.6770** (+8.97)	**0.7315** (+11.30)	**0.7641** (+13.49)
	USFTH	**0.8451** (+10.16)	**0.8790** (+11.23)	**0.8890** (+11.68)	**0.8963** (+10.01)	**0.6813** (+5.28)	**0.7726** (+9.82)	**0.7936** (+11.33)	**0.8057** (+9.53)	**0.5462** (+3.53)	**0.6568** (+8.85)	**0.7157** (+12.57)	**0.7558** (+14.51)
	ENHash	**0.8730** (+8.44)	**0.8935** (+2.70)	**0.9054** (+2.12)	**0.9031** (+1.78)	**0.7613** (+2.23)	**0.7891** (+2.70)	**0.8040** (+3.38)	**0.8094** (+2.83)	**0.6826** (+1.35)	**0.7262** (+2.99)	**0.7561** (+4.47)	**0.7840** (+5.23)
	HINT	**0.8695** (+5.42)	**0.8789** (+7.43)	**0.8980** (+8.71)	**0.8960** (+9.57)	**0.7510** (+3.46)	**0.7729** (+5.38)	**0.7940** (+8.18)	**0.8025** (+9.70)	**0.6238** (+13.42)	**0.6563** (+12.44)	**0.7233** (+21.28)	**0.7630** (+21.71)
	UWMCH	**0.8576** (+9.65)	**0.8730** (+10.39)	**0.8917** (+8.35)	**0.8931** (+5.35)	**0.7008** (+8.08)	**0.7716** (+10.84)	**0.7972** (+8.93)	**0.8137** (+5.39)	**0.6355** (+5.45)	**0.7034** (+7.77)	**0.7509** (+6.79)	**0.7818** (+4.11)
T→I	HEH	**0.8342** (+7.77)	**0.8431** (+6.09)	**0.8671** (+5.70)	**0.8671** (+5.12)	**0.7146** (+5.50)	**0.7419** (+6.04)	**0.7614** (+6.77)	**0.7746** (+7.96)	**0.6071** (+6.51)	**0.6611** (+9.58)	**0.7155** (+12.32)	**0.7406** (+15.51)
	USFTH	**0.7985** (+10.80)	**0.8340** (+10.85)	**0.8531** (+10.68)	**0.8611** (+9.78)	**0.6172** (+7.66)	**0.7320** (+12.68)	**0.7544** (+9.26)	**0.7716** (+10.07)	**0.5405** (+3.44)	**0.6378** (+8.59)	**0.7052** (+12.61)	**0.7411** (+13.27)
	ENHash	**0.8237** (+8.08)	**0.8581** (+1.57)	**0.8634** (+2.05)	**0.8691** (+1.44)	**0.7338** (+1.55)	**0.7551** (+2.09)	**0.7717** (+2.26)	**0.7855** (+2.15)	**0.6661** (+1.48)	**0.7091** (+3.22)	**0.7421** (+4.13)	**0.7661** (+4.23)
	HINT	**0.8324** (+4.00)	**0.8540** (+5.90)	**0.8644** (+6.25)	**0.8686** (+8.02)	**0.7199** (+3.06)	**0.7463** (+6.71)	**0.7657** (+7.32)	**0.7707** (+6.99)	**0.5974** (+12.38)	**0.6447** (+11.04)	**0.7090** (+18.97)	**0.7452** (+20.25)
	UWMCH	**0.8237** (+7.38)	**0.8394** (+7.97)	**0.8550** (+6.51)	**0.8556** (+3.85)	**0.6866** (+7.04)	**0.7484** (+9.64)	**0.7752** (+7.71)	**0.7765** (+5.08)	**0.6323** (+4.65)	**0.6935** (+6.80)	**0.7367** (+5.75)	**0.7665** (+3.80)

Here $\mathrm{BCEWithLogits}$ denotes binary cross-entropy applied directly to logits. The shared graph target transfers the same residual relational structure into both modalities, while separate predictors accommodate their heterogeneous feature spaces. Only the lightweight residual heads are optimized, leaving the original hash learner unchanged.

For an unseen image or text sample, the residual bits are obtained by thresholding the corresponding logits at zero, equivalently yielding $\widehat{\mathbf{z}}_i^v = \mathrm{sign}(g_v(\mathbf{f}_i^v))$ and $\widehat{\mathbf{z}}_i^t = \mathrm{sign}(g_t(\mathbf{f}_i^t))$.

Fixed-Budget Capacity Reallocation. ReCapHash does not append the recovered residual bits to the original representation. Instead, all effective dimensions indexed by $\mathcal{C}$ are preserved, while the K dimensions identified as reclaimable are reassigned to the recovered residual structure. For modality $* \in \{v, t\}$, the final representation can be written, up to a common permutation of hash dimensions, as

$$\widetilde{\mathbf{b}}_i^* = \left[\mathbf{b}_{i,\mathcal{C}}^*; \widehat{\mathbf{z}}_i^*\right] \in \{-1, +1\}^B. \tag{17}$$

Equivalently, the recovered residual dimensions can be written back to the original coordinates indexed by $\mathcal{R}$. Since the same coordinate organization is used for both modalities, the representation remains compatible with standard Hamming retrieval.

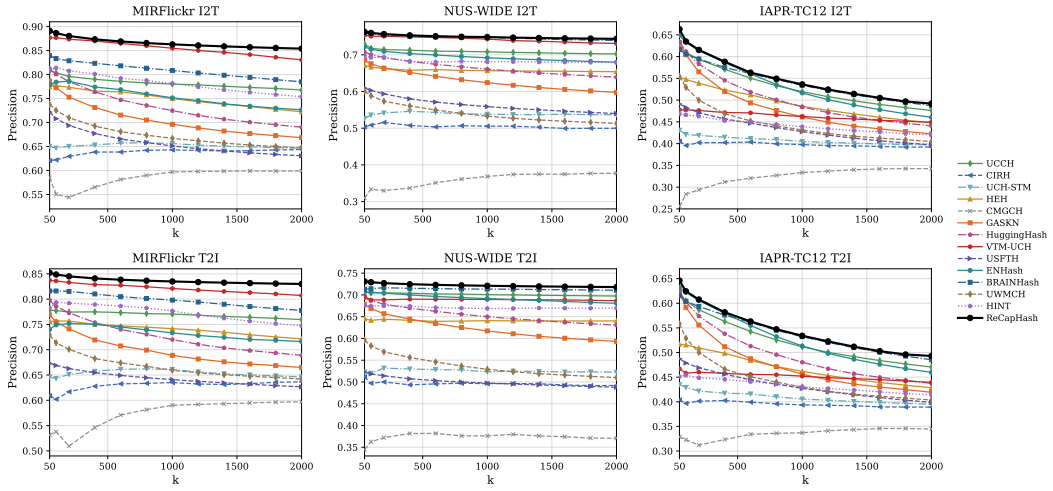


Figure 3: Top-k precision curves of ReCapHash and baselines on three datasets with 16-bit codes.

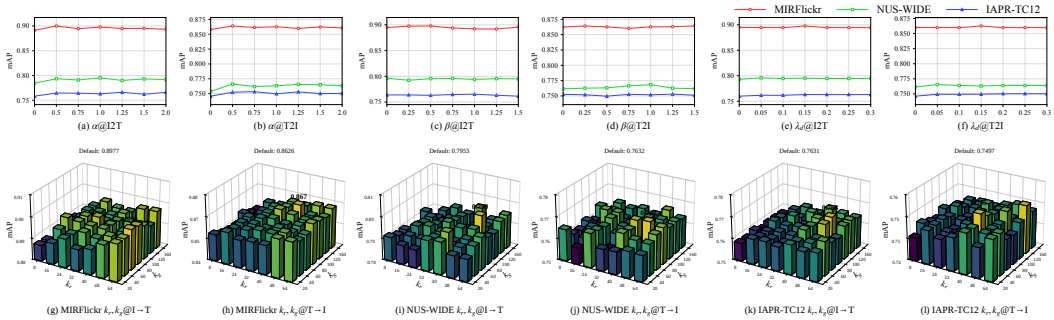


Figure 4: Parameter sensitivity analysis of α, β, λ_d, k_r, and k_g on the MIRFlickr, NUS-WIDE, and IAPR-TC12 datasets with a code length of 32 bits.

5 Experiment

5.1 Experimental Settings

❶ **Datasets.** We conduct experiments on three widely used cross-modal retrieval benchmarks: MIRFlickr-25K (Huiskes & Lew, 2008), NUS-WIDE (Chua et al., 2009), and IAPR-TC12 (Escalante et al., 2010). Details are provided in Appendix D.1. ❷ **Baselines.** To comprehensively evaluate the retrieval performance of ReCapHash, we compare it with 13 representative cross-modal hashing methods, including UCCH (Hu et al., 2022), CIRH (Zhu et al., 2022), UCH-STM (Tu et al., 2023), HEH (Zhong et al., 2023), CMGCH (Liang et al., 2024), GASKN (Yu et al., 2024), HuggingHash (Wang et al., 2024), VTM-UCH (Fan & Cao, 2025), USFTH (Yang et al., 2025), ENHash (Fu et al., 2026b), BRAINHash (Fu et al., 2026a), UWMCH (Yang et al., 2026b), and HINT (Xiao et al., 2026). Details are provided in Appendix D.2. ❸ **Implementation Details.** The hyperparameters α, β, λ_d, k_r, and k_g are fixed across all three datasets as $\alpha = 1.0$, $\beta = 0.5$, $\lambda_d = 0.15$, $k_r = 32$, and $k_g = 80$. All baselines are evaluated under the same protocol for fair comparison.

5.2 Results

Main Results. As shown in Table 1, ReCapHash consistently outperforms all baselines across the three benchmarks and different code lengths. Key observations are: ❶ ReCapHash achieves the

Table 3: Ablation study of ReCapHash and its design variants on three datasets.

Task	Method	MIRFlickr				NUS-WIDE				IAPR-TC12			
		16 bits	32 bits	64 bits	128 bits	16 bits	32 bits	64 bits	128 bits	16 bits	32 bits	64 bits	128 bits
I→T	Fixed-25%	0.8800	0.8905	0.9034	0.8979	0.7568	0.7933	0.8149	0.8127	0.6759	0.7566	0.7861	0.8020
	Random Reclaim	0.8724	0.8796	0.8923	0.8880	0.7568	0.7828	0.8015	0.8004	0.6590	0.7493	0.7667	0.7765
	Semantic-only	0.8808	0.8910	0.8992	0.8986	0.7569	0.7846	0.8080	0.8075	0.6647	0.7583	0.7797	0.7976
	Random Relation	0.8089	0.8144	0.8390	0.8244	0.5786	0.6616	0.7256	0.7430	0.5303	0.7037	0.7044	0.7214
	Local Recovery	0.8354	0.8358	0.8559	0.8575	0.7205	0.7725	0.7895	0.8002	0.6538	0.7536	0.7678	0.7766
	ReCapHash	**0.8871**	**0.8977**	**0.9044**	**0.9051**	**0.7719**	**0.7953**	**0.8152**	**0.8131**	**0.6764**	**0.7631**	**0.7863**	**0.8042**
T→I	Fixed-25%	0.8496	0.8590	0.8594	0.8691	0.7366	0.7617	0.7786	0.7815	0.6653	0.7441	0.7756	0.7877
	Random Reclaim	0.8376	0.8466	0.8501	0.8559	0.7302	0.7517	0.7673	0.7662	0.6548	0.7300	0.7641	0.7709
	Semantic-only	0.8466	0.8579	0.8597	0.8658	0.7271	0.7538	0.7751	0.7828	0.6569	0.7457	0.7717	0.7851
	Random Relation	0.7955	0.8033	0.8168	0.8159	0.5736	0.6461	0.7039	0.7215	0.5366	0.6995	0.7079	0.7259
	Local Recovery	0.8172	0.8299	0.8405	0.8457	0.7139	0.7571	0.7791	0.7816	0.6666	0.7490	0.7711	0.7816
	ReCapHash	**0.8510**	**0.8626**	**0.8634**	**0.8701**	**0.7404**	**0.7632**	**0.7813**	**0.7881**	**0.6674**	**0.7497**	**0.7772**	**0.7902**

best performance in all 24 settings across datasets, code lengths, and retrieval directions; ❷ the gains are especially clear on MIRFlickr and IAPR-TC12; ❸ consistent improvements are obtained for both I→T and T→I retrieval; and ❹ the advantage persists from compact to longer hash codes, confirming the effectiveness of residual capacity reallocation across bit budgets. Figure 3 further shows that ReCapHash maintains the highest Top-k precision across retrieval depths on all datasets.

Plug-in Generality. As shown in Table 2, we evaluate ReCapHash as a plug-in on five representative base hashing methods, including HEH (Zhong et al., 2023), USFTH (Yang et al., 2025), ENHash (Fu et al., 2026b), HINT (Xiao et al., 2026), and UWMCH (Yang et al., 2026b). ReCapHash consistently improves each evaluated method across datasets, code lengths, and retrieval directions, indicating that its effectiveness is not tied to a particular base hashing method. These consistent improvements demonstrate the plug-in generality of ReCapHash under a fixed hash budget.

Ablation Study. Table 3 evaluates five variants of ReCapHash. Fixed-25% uses a fixed 25% reclaiming ratio while selecting the lowest-utility bits; Random Reclaim keeps the adaptive reclaiming size but randomly chooses which positions to replace; Semantic-only removes the retained-core affinity term and mines relations solely from semantic affinity; Random Relation replaces the mined residual relations with random ones; and Local Recovery replaces global spectral recovery with a local strategy. ReCapHash achieves the best performance in all settings. Compared with Random Reclaim, utility-guided bit selection yields consistent gains, confirming that reclaiming low-utility coordinates is preferable to arbitrary replacement. Fixed-25% remains competitive but consistently underperforms ReCapHash, supporting adaptive capacity determination. Semantic-only also degrades performance, showing the benefit of conditioning residual relation discovery on what is already represented by the retained core. The larger drops of Random Relation and Local Recovery further demonstrate the importance of meaningful residual relation mining and global residual recovery.

Parameter Sensitivity. Figure 4 studies the sensitivity of ReCapHash to α, β, λ_d, k_r, and k_g using 32-bit codes. Overall, ReCapHash remains stable over broad parameter ranges on all three datasets and both retrieval directions. The performance varies only moderately with α, β, and λ_d, indicating that the residual suppression, reciprocal weighting, and diversity regularization are not sensitive to precise parameter choices. The joint analysis of k_r and k_g exhibits broad high-performing regions rather than isolated optima, showing robustness to the construction density of the residual graph.

6 Conclusion

This paper investigates fixed-budget capacity utilization in unsupervised cross-modal hashing. We propose ReCapHash, which identifies reclaimable low-utility hash dimensions through bidirectional marginal utility and preserves the remaining dimensions as an effective core. By mining core-conditioned residual relations, organizing them into a paired residual graph, and recovering complementary residual bits through global spectral recovery and cross-modal distillation, ReCapHash reallocates existing capacity without increasing the original code length. Extensive experiments demonstrate the effectiveness of ReCapHash across benchmark datasets, code lengths, and retrieval directions. Consistent gains over diverse base hashing methods further validate its plug-in generality.

AI USE STATEMENT

We used generative AI tools to assist with language editing and polishing, as well as literature retrieval and discovery. All AI-assisted content was reviewed by the authors, who take full responsibility for the final content of this work.

REFERENCES

Mothilal Asokan, Kebin Wu, and Fatima Albreiki. Finelip: Extending clip's reach via fine-grained alignment with longer text inputs. In *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 14495–14504. IEEE, 2025.

Davide Caffagni, Sara Sarto, Marcella Cornia, Lorenzo Baraldi, and Rita Cucchiara. Recurrence-enhanced vision-and-language transformers for robust multimodal document retrieval. In *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 9286–9295. IEEE, 2025.

Yuan Cao, Zifan Liu, Weikang Gao, Jie Gui, and Yanwei Yu. Long-tailed approaching cross-modal hashing with multi-expert collaborative learning. *IEEE Transactions on Image Processing*, 2026.

Jiali Chen, Ruitao Pu, Dezhong Peng, Xiaomin Song, Yingke Chen, and Yuan Sun. External guidance incomplete cross-modal hashing. *IEEE Transactions on Image Processing*, 2026.

Tat-Seng Chua, Jinhui Tang, Richang Hong, Haojie Li, Zhiping Luo, and Yantao Zheng. 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*, pp. 1–9, 2009.

Yuhao Cui, Xinxing Zu, Wenhua Zhang, Zhongzhou Zhao, and Jinyang Gao. Incorporating dense knowledge alignment into unified multimodal representation models. In *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 29733–29743. IEEE, 2025.

Siyuan Duan, Yuan Sun, Dezhong Peng, Zheng Liu, Xiaomin Song, and Peng Hu. Fuzzy multimodal learning for trusted cross-modal retrieval. In *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 20747–20756. IEEE, 2025.

Hugo Jair Escalante, Carlos A Hernández, Jesus A Gonzalez, Aurelio López-López, Manuel Montes, Eduardo F Morales, L Enrique Sucar, Luis Villasenor, and Michael Grubinger. The segmented and annotated iapr tc-12 benchmark. *Computer vision and image understanding*, 114 (4):419–428, 2010.

Haozhi Fan and Yuan Cao. Vision-guided text mining for unsupervised cross-modal hashing with community similarity quantization. In *Proceedings of the AAAI conference on artificial intelligence*, volume 39, pp. 2843–2851, 2025.

Wentao Fan, Chao Zhang, Chunlin Chen, and Huaxiong Li. Online cross-modal hashing with expanding label space. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pp. 21002–21010, 2026.

Hao Fu, Guanghua Gu, Yunchao Wei, and Yao Zhao. Brainhash: Brain-inspired region-aligned interaction network for unsupervised cross-modal hashing. *IEEE Transactions on Image Processing*, 2026a.

Hao Fu, Zebing Yao, Chuangchuang Tan, and Guanghua Gu. Enhash: Error notebook-guided fine-grained learning for unsupervised cross-modal hashing. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pp. 21136–21144, 2026b.

Shuo Han, Qibing Qin, Kezhen Xie, Wenfeng Zhang, and Lei Huang. Polysemic semantic instance network for cross-modal hashing. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pp. 4592–4600, 2026.

Peng Hu, Hongyuan Zhu, Jie Lin, Dezhong Peng, Yin-Ping Zhao, and Xi Peng. Unsupervised contrastive cross-modal hashing. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 45(3):3877–3889, 2022.

Zhikai Hu, Yiu-Ming Cheung, Mengke Li, and Weichao Lan. Cross-modal hashing method with properties of hamming space: A new perspective. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 46(12):7636–7650, 2024.

Mark J Huiskes and Michael S Lew. The mir flickr retrieval evaluation. In *Proceedings of the 1st ACM international conference on Multimedia information retrieval*, pp. 39–43, 2008.

Young Kyun Jang and Ser-nam Lim. Towards cross-modal backward-compatible representation learning for vision-language models. In *2025 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 1783–1792. IEEE, 2025.

Kaihang Jiang, Wai Keung Wong, Xiaozhao Fang, Weijun Sun, Guoxu Zhou, Shengli Xie, and Xiaochun Cao. An efficient regenerated cross-modal hashing: Improving existing hash codes with the arbitrary length. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2026.

T P Kalyan, Piyush S Pasi, Sahil N Dharod, Azeem A Motiwala, Preethi Jyothi, Aditi Chaudhary, and Krishna Srinivasan. Wikido: A new benchmark evaluating cross-modal retrieval for vision-language models. *Advances in Neural Information Processing Systems*, 37:140812–140827, 2024.

Xiao Kang, Xingbo Liu, Xuening Zhang, Wen Xue, Xiushan Nie, and Yilong Yin. Semi-supervised online cross-modal hashing. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pp. 17770–17778, 2025.

Mingjin Kuai, Jun Long, and Zhan Yang. Statistical model-driven similarity hashing: bridging modalities for efficient unsupervised retrieval. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pp. 11977–11986, 2025.

Jiaxing Li, Lin Jiang, Zeqi Ma, Kaihang Jiang, Xiaozhao Fang, and Jie Wen. Lightweight contrastive distilled hashing for online cross-modal retrieval. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pp. 4779–4787, 2025a.

Yinan Li, Jun Long, and Zhan Yang. Asymmetric cross-modal hashing based on formal concept analysis. In *Proceedings of the AAAI conference on artificial intelligence*, volume 39, pp. 1392–1401, 2025b.

Yuan Li, Liangli Zhen, Yuan Sun, Dezhong Peng, Xi Peng, and Peng Hu. Deep evidential hashing for trustworthy cross-modal retrieval. In *Proceedings of the AAAI conference on artificial intelligence*, volume 39, pp. 18566–18574, 2025c.

Meiyu Liang, Junping Du, Zhengyang Liang, Yongwang Xing, Wei Huang, and Zhe Xue. Self-supervised multi-modal knowledge graph contrastive hashing for cross-modal search. In *Proceedings of the AAAI conference on artificial intelligence*, volume 38, pp. 13744–13753, 2024.

Zengrong Lin, Zheng Wang, Tianwen Qian, Pan Mu, Sixian Chan, and Cong Bai. Neighborretr: Balancing hub centrality in cross-modal retrieval. In *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 9263–9273. IEEE, 2025.

Kaiming Liu, Yunhong Gong, Yu Cao, Zhenwen Ren, Dezhong Peng, and Yuan Sun. Dual semantic fusion hashing for multi-label cross-modal retrieval. In *Ijcai*, pp. 4569–4577, 2024.

Wenhao Liu, Tianshi Wang, Fengling Li, Jingjing Li, Guoli Yang, and Lei Zhu. Do more with less: Capacity-aware selective hashing for continual cross-modal retrieval. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2026.

Xianglong Liu, Junfeng He, Bo Lang, and Shih-Fu Chang. Hash bit selection: a unified solution for selection problems in hashing. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 1570–1577, 2013.

Xiaoqing Liu, Zhiwen Yu, Kaixiang Yang, Jun Yu, Huanqiang Zeng, and CL Philip Chen. Enhanced cross-modal hashing via hybrid distillation and structural refinement. *IEEE Transactions on Image Processing*, 2025.

Junyu Luo, Jinsheng Huang, Yang Xu, Lutong Zou, Xiao Luo, Bohan Wu, Yifan Wang, Wei Ju, and Ming Zhang. Conformalized hierarchical calibration for uncertainty-aware adaptive hashing. In *International Conference on Learning Representations*, volume 2026, pp. 105223–105246, 2026.

Likang Peng, Chao Su, Wenyuan Wu, Yuan Sun, Dezhong Peng, Xi Peng, and Xu Wang. Semantic-consistent bidirectional contrastive hashing for noisy multi-label cross-modal retrieval. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pp. 24811–24819, 2026.

Ruitao Pu, Yuan Sun, Yang Qin, Zhenwen Ren, Xiaomin Song, Huiming Zheng, and Dezhong Peng. Robust self-paced hashing for cross-modal retrieval with noisy labels. In *Proceedings of the AAAI conference on artificial intelligence*, volume 39, pp. 19969–19977, 2025.

Qibing Qin, Lei Wu, Wenfeng Zhang, Huihui Zhang, Lei Huang, and Jie Nie. Deep semantic tuplet-based hashing by hypergraph modeling for cross-modal retrieval. *IEEE Transactions on Multimedia*, 2025.

Chao Su, Yanan Li, Xu Wang, Yingke Chen, Huiming Zheng, Dezhong Peng, and Yuan Sun. Ambiguity-tolerant cross-modal hashing with partial labels. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pp. 25636–25644, 2026a.

Chao Su, Likang Peng, Yuan Sun, Dezhong Peng, Xi Peng, and Xu Wang. Neighbor-aware contrastive disambiguation for cross-modal hashing with redundant annotations. *Advances in Neural Information Processing Systems*, 38:108581–108603, 2026b.

Hao Sun, Yadong Huo, Qibing Qin, Wenfeng Zhang, and Lei Huang. Intra-class distribution-guided generative hashing with neighbor refinement for cross-modal retrieval. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 2671–2681, June 2026.

Yuan Sun, Kaiming Liu, Yongxiang Li, Zhenwen Ren, Jian Dai, and Dezhong Peng. Distribution consistency guided hashing for cross-modal retrieval. In *Proceedings of the 32nd ACM International Conference on Multimedia*, pp. 5623–5632, 2024.

Wentao Tan, Fengling Li, Lei Zhu, Weili Guan, Jingjing Li, Zhiyong Cheng, and Heng Tao Shen. Dynamic bit-wise semantic transformer hashing for multi-modal retrieval. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2025.

Ziang Tan, Weitao An, and Erkun Yang. Meta-guided sample reweighting for robust cross-modal hashing retrieval with noisy labels. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pp. 9350–9358, 2026.

Junfeng Tu, Xueliang Liu, Zhen Huang, Yanbin Hao, Richang Hong, and Meng Wang. Cross-modal hashing via diverse instances matching. *IEEE Transactions on Image Processing*, 34:2737–2749, 2025.

Rong-Cheng Tu, Xian-Ling Mao, Qinghong Lin, Wenjin Ji, Weize Qin, Wei Wei, and Heyan Huang. Unsupervised cross-modal hashing via semantic text mining. *IEEE Transactions on Multimedia*, 25:8946–8957, 2023.

Jinpeng Wang, Ziyun Zeng, Bin Chen, Yuting Wang, Dongliang Liao, Gongfu Li, Yiru Wang, and Shu-Tao Xia. Hugs bring double benefits: Unsupervised cross-modal hashing with multi-granularity aligned transformers. *International Journal of Computer Vision*, 132(8):2765–2797, 2024.

Min Wang, Wengang Zhou, Xin Yao, and Houqiang Li. Adaptive bit selection for scalable deep hashing. *IEEE Transactions on Image Processing*, 34:1048–1059, 2025.

Zhiping Xiao, Junyu Luo, Hang Zhou, Yusheng Zhao, Xiao Luo, Pengyun Wang, Wei Ju, Siyu Heng, and Ming Zhang. Hierarchical encoding tree with modality mixup for cross-modal hashing. In *International Conference on Learning Representations*, volume 2026, pp. 126617–126641, 2026.

Fan Yang, Tongxuan Pei, Yuanzhi Zhao, and Yudong Zhao. Dssl-hash: Dynamic semantic structure learning for unsupervised cross-modal hashing. In *Proceedings of the Thirty-Fifth International Joint Conference on Artificial Intelligence and the 29th European Conference on Artificial Intelligence, IJCAI/ECAI 2026, Bremen, Germany, 15-21 August 2026*, pp. 1965–1973. ijcai.org, 2026a. doi: 10.24963/IJCAI.2026/219. URL https://doi.org/10.24963/ijcai.2026/219.

Fan Yang, Yuanzhi Zhao, Haimei Zhao, Yudong Zhao, and Haikun Xu. Mask to align, weight to disambiguate: Reliable unsupervised cross-modal hashing with masked-weight contrast. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 30151–30161, June 2026b.

Zhan Yang, Binghong Chen, Jiajun Tang, and Yinan Li. Unsupervised similarity-fusion transformer hashing for multimodal retrieval. In *Proceedings of the 33rd ACM International Conference on Multimedia*, pp. 5932–5941, 2025.

Zhan Yang, Yiran Liu, Youyuan Huang, and Yinan Li. Stationary and clustering transformer hashing for cross-modal retrieval. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pp. 27728–27737, 2026c.

Yang Yu, Meiyu Liang, Mengran Yin, Kangkang Lu, Junping Du, and Zhe Xue. Unsupervised multimodal graph contrastive semantic anchor space dynamic knowledge distillation network for cross-media hash retrieval. In *2024 IEEE 40th International Conference on Data Engineering (ICDE)*, pp. 4699–4708. IEEE, 2024.

Zhongqing Yu, Xin Liu, Yiu-ming Cheung, Zhikai Hu, Wentao Fan, and Pan Zhou. Drfgd: Disentangled representation-focused generative defense for attack-tolerant cross-modal hashing. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pp. 16226–16234, 2026.

Donglin Zhang, Zhikai Hu, Xiao-Jun Wu, and Josef Kittler. Supervised discrete cross-modal hashing with exploiting semantic correlations. *Pattern Recognition*, pp. 113727, 2026a.

Shuang Zhang, Yue Wu, Lei Shi, Huilong Jin, Feifei Kou, Pengfei Zhang, Mingying Xu, and Pengtao Lv. Adaptive graph attention based discrete hashing for incomplete cross-modal retrieval. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pp. 28382–28390, 2026b.

Shuang Zhang, Yue Wu, Lei Shi, Yingxue Zhang, Feifei Kou, Huilong Jin, Pengfei Zhang, Meiyu Liang, and Mingying Xu. Dynamic masking and auxiliary hash learning for enhanced cross-modal retrieval. *Advances in Neural Information Processing Systems*, 38:35485–35506, 2026c.

Yuanzhi Zhao, Fan Yang, Yudong Zhao, and Xiaoyu Li. Udch: Unsupervised dynamic weighted cluster-cooperative hashing for cross-modal retreival. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pp. 13297–13304, 2026.

Fangming Zhong, Chenglong Chu, Zijie Zhu, and Zhikui Chen. Hypergraph-enhanced hashing for unsupervised cross-modal retrieval via robust similarity guidance. In *Proceedings of the 31st ACM international conference on multimedia*, pp. 3517–3527, 2023.

Lei Zhu, Xize Wu, Jingjing Li, Zheng Zhang, Weili Guan, and Heng Tao Shen. Work together: Correlation-identity reconstruction hashing for unsupervised cross-modal retrieval. *IEEE Transactions on Knowledge and Data Engineering*, 35(9):8838–8851, 2022.

Qiang Zou, Shuli Cheng, and Jiayi Chen. Prompthash: Affinity-prompted collaborative cross-modal learning for adaptive hashing retrieval. In *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 19649–19658. IEEE, 2025.

Ruifan Zuo, Chaoqun Zheng, Fengling Li, Tianshi Wang, Tian Gan, Wenpeng Lu, and Xiaoming Wu. Beyond fixed-length constraints: Variable-length hashing via adaptive bit-importance learning for multi-modal retrieval. *IEEE Transactions on Multimedia*, 2026.

A ALGORITHM

We summarize the complete procedure of ReCapHash in Algorithm 1.

Algorithm 1 ReCapHash

Require: Semantic features $\mathbf{F}^v, \mathbf{F}^t$; base hash codes $\mathbf{B}^v, \mathbf{B}^t \in \{-1, +1\}^{N \times B}$; k_p, k_n, k_s, k_r, k_g; α, β, λ_d

Ensure: Reallocated codes $\widetilde{\mathbf{B}}^v, \widetilde{\mathbf{B}}^t$

$\mathcal{B} \leftarrow \{1, \ldots, B\}$

Stage 1: Utility-guided capacity reclamation

Construct semantic neighborhoods $\{\mathcal{P}_i^t, \mathcal{N}_i^t, \mathcal{P}_i^v, \mathcal{N}_i^v\}_{i=1}^N$

for $b = 1$ to B **do**

  Compute $u_b^{v \to t}$ and $u_b^{t \to v}$ by Eq. (4)

  $u_b \leftarrow (u_b^{v \to t} + u_b^{t \to v})/2$

end for

$\pi \leftarrow \mathrm{argsort}(\{u_b\}_{b=1}^B)$

Determine the adaptive split K by Eq. (5)

$\mathcal{R} \leftarrow \{\pi_1, \ldots, \pi_K\}$; $\mathcal{C} \leftarrow \mathcal{B} \setminus \mathcal{R}$

Stage 2: Core-conditioned residual graph

Compute $\mathbf{S}^{\mathrm{sem}}$ and $\mathbf{S}^{\mathrm{core}}$ using $\mathcal{C}$

Row-standardize $\mathbf{S}^{\mathrm{sem}}$ and $\mathbf{S}^{\mathrm{core}}$

for $i = 1$ to N **do**

  Compute $\{r_{ij}^{v \to t}\}_j$ and $\{r_{ij}^{t \to v}\}_j$ by Eq. (7)

  $\mathcal{N}_i^{v \to t} \leftarrow$ top-k_r residuals within top-k_s semantic candidates

  $\mathcal{N}_i^{t \to v} \leftarrow$ top-k_r residuals within top-k_s semantic candidates

end for

Construct $\mathbf{A}$ by Eq. (8)

$\widetilde{A}_{ij} \leftarrow A_{ij}(1 + \beta \mathbb{I}_{ij}^{\mathrm{rec}})$ for all retained relations

$\mathbf{W} \leftarrow (\widetilde{\mathbf{A}} + \widetilde{\mathbf{A}}^\top)/2$

Retain top-k_g affinities per row and set $\mathbf{W} \leftarrow \max(\mathbf{W}, \mathbf{W}^\top)$

Stage 3: Global residual-bit recovery

$\mathbf{D} \leftarrow \mathrm{diag}(\mathbf{W}\mathbf{1})$; $\mathbf{L} \leftarrow \mathbf{D} - \mathbf{W}$

Obtain $\mathbf{Y}^* \in \mathbb{R}^{N \times K}$ by Eq. (11)

$\mathcal{Q} \leftarrow \emptyset$

for each ITQ restart **do**

  Initialize orthogonal rotation $\mathbf{R}$

  repeat

    $\mathbf{Z} \leftarrow \mathrm{sign}(\mathbf{Y}^*\mathbf{R})$

    Update $\mathbf{R}$ by orthogonal Procrustes

  until convergence

  $\mathcal{Q} \leftarrow \mathcal{Q} \cup \{\mathbf{Z}\}$

end for

$\mathbf{Z}^* \leftarrow \arg\max_{\mathbf{Z} \in \mathcal{Q}} [\mathcal{G}(\mathbf{Z}; \mathbf{W}) - \lambda_d \rho_{\mathrm{bit}}(\mathbf{Z})]$

Stage 4: Residual distillation and reallocation

$\mathbf{Y}^{\mathrm{tar}} \leftarrow (\mathbf{Z}^* + 1)/2$

Initialize residual heads g_v and g_t

repeat

  Update g_v, g_t by minimizing $\mathcal{L}_{\mathrm{dist}}$ in Eq. (16)

until convergence

$\widehat{\mathbf{Z}}^v \leftarrow \mathrm{sign}(g_v(\mathbf{F}^v))$

$\widehat{\mathbf{Z}}^t \leftarrow \mathrm{sign}(g_t(\mathbf{F}^t))$

$\widetilde{\mathbf{B}}^v \leftarrow \mathbf{B}^v$; $\widetilde{\mathbf{B}}^t \leftarrow \mathbf{B}^t$

$\widetilde{\mathbf{B}}_{:,\mathcal{R}}^v \leftarrow \widehat{\mathbf{Z}}^v$; $\widetilde{\mathbf{B}}_{:,\mathcal{R}}^t \leftarrow \widehat{\mathbf{Z}}^t$

return $\widetilde{\mathbf{B}}^v, \widetilde{\mathbf{B}}^t$

B Theoretical Properties

B.1 Spectral Solution of Global Residual Recovery

Proposition 1. Assume that the residual graph is connected. Let $0 = \lambda_1 < \lambda_2 \leq \cdots \leq \lambda_N$ be the generalized eigenvalues satisfying $\mathbf{L}\mathbf{y}_i = \lambda_i \mathbf{D}\mathbf{y}_i$, with $\mathbf{D}$-orthonormal eigenvectors. Then

$$\mathbf{Y}^* = [\mathbf{y}_2, \ldots, \mathbf{y}_{K+1}], \qquad \mathrm{Tr}\big((\mathbf{Y}^*)^\top \mathbf{L}\mathbf{Y}^*\big) = \sum_{i=2}^{K+1} \lambda_i.$$

Proof. Let $\mathbf{L}_\mathrm{n} = \mathbf{D}^{-1/2}\mathbf{L}\mathbf{D}^{-1/2}$ and $\mathbf{U} = \mathbf{D}^{1/2}\mathbf{Y}$. Substituting $\mathbf{Y} = \mathbf{D}^{-1/2}\mathbf{U}$ into Eq. (11) yields

$$\min_{\mathbf{U}} \mathrm{Tr}(\mathbf{U}^\top \mathbf{L}_\mathrm{n} \mathbf{U}), \qquad \text{s.t. } \mathbf{U}^\top\mathbf{U} = \mathbf{I}, \quad \mathbf{U}^\top \mathbf{D}^{1/2}\mathbf{1} = 0.$$

Since the residual graph is connected, $\mathbf{L}_\mathrm{n}$ is symmetric positive semidefinite and its smallest eigenvalue is $\lambda_1 = 0$, with eigenvector proportional to $\mathbf{D}^{1/2}\mathbf{1}$. The second constraint therefore excludes the trivial direction. By the Rayleigh–Ritz theorem, the minimum is attained by the K eigenvectors $\mathbf{u}_2, \ldots, \mathbf{u}_{K+1}$ associated with the smallest non-trivial eigenvalues. Setting $\mathbf{y}_i = \mathbf{D}^{-1/2}\mathbf{u}_i$ gives $\mathbf{L}\mathbf{y}_i = \lambda_i \mathbf{D}\mathbf{y}_i$, and hence the stated solution and objective value. □

B.2 Core-Margin Preservation under Fixed-Budget Reallocation

Proposition 2. Let $\mathbf{b}_i^* = [\mathbf{b}_{i,\mathcal{C}}^*; \mathbf{b}_{i,\mathcal{R}}^*]$ denote the original B-bit code and $\tilde{\mathbf{b}}_i^* = [\mathbf{b}_{i,\mathcal{C}}^*; \hat{\mathbf{z}}_i^*]$ the reallocated code in Eq. (17), where $|\mathcal{R}| = K$ and $* \in \{v, t\}$. For any cross-modal pair (i, j),

$$\left| d_H(\tilde{\mathbf{b}}_i^v, \tilde{\mathbf{b}}_j^t) - d_H(\mathbf{b}_i^v, \mathbf{b}_j^t) \right| \leq K.$$

Moreover, for any query q, positive sample p, and negative sample n, if the preserved core satisfies

$$d_H(\mathbf{b}_{q,\mathcal{C}}, \mathbf{b}_{n,\mathcal{C}}) - d_H(\mathbf{b}_{q,\mathcal{C}}, \mathbf{b}_{p,\mathcal{C}}) > K,$$

then the corresponding ranking is preserved after reallocation, i.e., $d_H(\tilde{\mathbf{b}}_q, \tilde{\mathbf{b}}_p) < d_H(\tilde{\mathbf{b}}_q, \tilde{\mathbf{b}}_n)$.

Proof. Because ReCapHash leaves all dimensions in $\mathcal{C}$ unchanged, the original and reallocated Hamming distances decompose as

$$d_H(\mathbf{b}_i^v, \mathbf{b}_j^t) = d_H(\mathbf{b}_{i,\mathcal{C}}^v, \mathbf{b}_{j,\mathcal{C}}^t) + d_H(\mathbf{b}_{i,\mathcal{R}}^v, \mathbf{b}_{j,\mathcal{R}}^t), \qquad d_H(\tilde{\mathbf{b}}_i^v, \tilde{\mathbf{b}}_j^t) = d_H(\mathbf{b}_{i,\mathcal{C}}^v, \mathbf{b}_{j,\mathcal{C}}^t) + d_H(\hat{\mathbf{z}}_i^v, \hat{\mathbf{z}}_j^t).$$

Both Hamming distances over the K replaced dimensions lie in $[0, K]$, and therefore their difference has magnitude at most K, proving the first claim.

For the ranking result, define the core margin $m_\mathcal{C} = d_H(\mathbf{b}_{q,\mathcal{C}}, \mathbf{b}_{n,\mathcal{C}}) - d_H(\mathbf{b}_{q,\mathcal{C}}, \mathbf{b}_{p,\mathcal{C}})$. After reallocation,

$$d_H(\tilde{\mathbf{b}}_q, \tilde{\mathbf{b}}_n) - d_H(\tilde{\mathbf{b}}_q, \tilde{\mathbf{b}}_p) = m_\mathcal{C} + d_H(\hat{\mathbf{z}}_q, \hat{\mathbf{z}}_n) - d_H(\hat{\mathbf{z}}_q, \hat{\mathbf{z}}_p) \geq m_\mathcal{C} - K.$$

Hence, whenever $m_\mathcal{C} > K$, the final distance margin remains positive, which implies $d_H(\tilde{\mathbf{b}}_q, \tilde{\mathbf{b}}_p) < d_H(\tilde{\mathbf{b}}_q, \tilde{\mathbf{b}}_n)$. □

B.3 Affine Invariance of Adaptive Capacity Splitting

Lemma 1. Let $\{u_b\}_{b=1}^B$ denote the bit utilities used in Eq. (5), and consider a positive affine transformation $\tilde{u}_b = a u_b + c$ with $a > 0$. Then the optimal split positions determined by Eq. (5) remain unchanged.

Proof. Since $a > 0$, the transformation preserves the ordering of the utilities, and therefore the permutation π obtained by sorting them in ascending order is unchanged. For any candidate split k, the transformed group means satisfy $\tilde{\mu}_-(k) = a\mu_-(k) + c$ and $\tilde{\mu}_+(k) = a\mu_+(k) + c$. Hence,

$$\frac{k(B-k)}{B^2} \left[\tilde{\mu}_-(k) - \tilde{\mu}_+(k)\right]^2 = a^2 \frac{k(B-k)}{B^2} \left[\mu_-(k) - \mu_+(k)\right]^2.$$

Since $a^2 > 0$ is independent of k, all candidate objectives are scaled by the same positive constant. Thus their ordering, and consequently the set of maximizers in Eq. (5), is unchanged. Therefore, under the same deterministic tie-breaking rule, the selected reclaimable capacity K is invariant to any positive affine transformation of the utility values. □

This property shows that adaptive capacity splitting depends on the relative structure of the utility profile rather than its absolute scale or offset.

C Additional Experimental Results

C.1 Residual Relation Validity and Complementarity

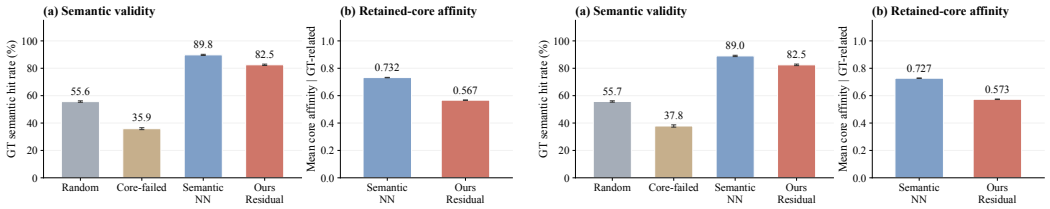


Figure 5: Quantitative evaluation of mined residual relations, with I→T shown on the left and T→I on the right. (a) Ground-truth (GT) semantic hit rates of different relation mining strategies. (b) Mean retained-core affinity of semantic nearest neighbors (NN) and our residual relations.

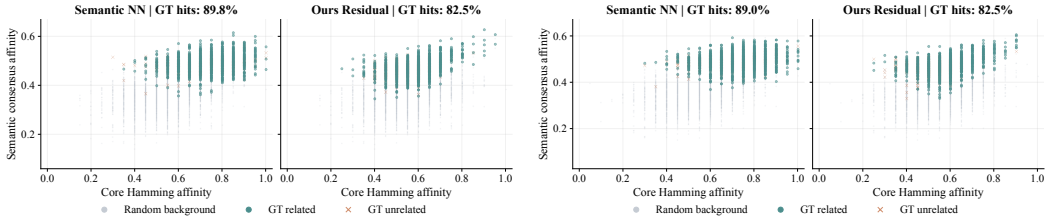


Figure 6: Semantic affinity versus retained-core Hamming affinity for semantic nearest neighbors and mined residual relations, with I→T shown on the left and T→I on the right. Markers distinguish ground-truth related and unrelated pairs.

To examine whether the mined residual relations remain semantically meaningful while providing information complementary to the retained core. As shown in Figure 5, our residual relations achieve high ground-truth semantic hit rates of 82.5% in both retrieval directions, substantially outperforming Random and Core-failed mining and approaching Semantic NN (89.8% for I→T and 89.0% for T→I). Meanwhile, their mean retained-core affinity is markedly lower than that of semantic nearest neighbors, decreasing from 0.732 to 0.567 for I→T and from 0.727 to 0.573 for T→I. These results indicate that ReCapHash preserves strong semantic validity while preferentially selecting relations that are insufficiently represented by the preserved core.

Figure 6 further provides distribution-level evidence for this behavior. Compared with semantic nearest neighbors, the mined residual relations shift toward lower core Hamming affinity while remaining concentrated in regions of relatively high semantic affinity. The consistent pattern across both retrieval directions confirms that core-conditioned residual mining does not merely rediscover generic semantic neighbors, but identifies semantically valid relations that complement the information already encoded by the retained core.

C.2 Validation of Bit Utility

To examine whether the proposed marginal utility u_b reliably reflects the contribution of individual hash dimensions, Figure 7 compares low-utility, random, and high-utility bit removal. Removing low-utility bits causes only minor performance variation in both I→T and T→I retrieval, even

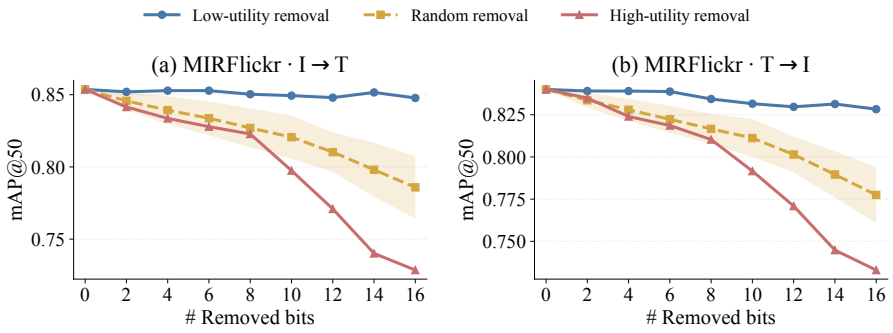


Figure 7: Bit utility validation on MIRFlickr with 32-bit base hash codes.

when up to half of the 32-bit code is removed. In contrast, random removal leads to a gradual decline, while removing high-utility bits causes substantially sharper degradation as more dimensions are discarded. The consistent ordering of the three strategies demonstrates that u_b effectively distinguishes dispensable dimensions from those carrying important retrieval information, directly supporting utility-guided capacity reclamation.

C.3 Adaptive Capacity Reclamation Analysis

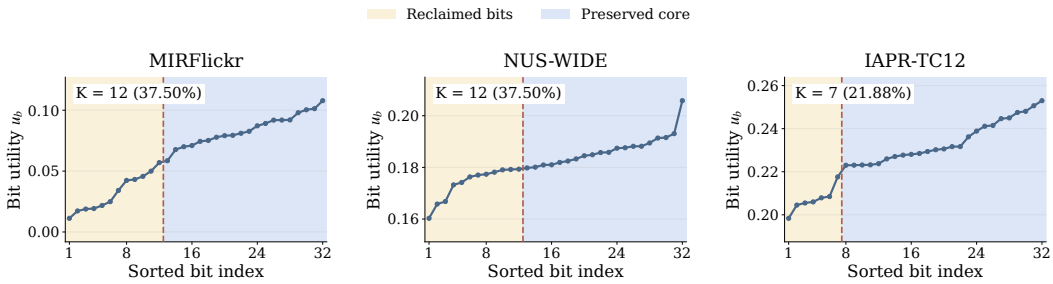


Figure 8: Utility profiles of 32-bit hash codes on MIRFlickr, NUS-WIDE, and IAPR-TC12. Bits are sorted in ascending order of utility u_b. The dashed line marks the adaptively selected reclamation boundary K, separating reclaimed bits from the preserved core.

To examine how ReCapHash determines the reclaimable capacity, Figure 8 visualizes the sorted bit-utility profiles of 32-bit base codes on the three datasets. The utilities are clearly non-uniform, indicating that different hash dimensions contribute unequally to retrieval. ReCapHash adaptively selects $K = 12$ bits (37.50%) on MIRFlickr and NUS-WIDE, but only $K = 7$ bits (21.88%) on IAPR-TC12. The varying boundaries show that the amount of reclaimable capacity depends on the learned representation rather than a predefined replacement ratio, supporting adaptive capacity determination under a fixed hash budget.

C.4 Residual Graph Structure Analysis

Table 4: Structural statistics of the residual relation graphs. Density is computed as $2|E|/(|V|(|V|-1))$. Reciprocal ratio denotes the fraction of directed residual evidence supported in the reverse direction before final graph symmetrization.

Dataset	$\lvert V\rvert$	$\lvert E\rvert$	Average Degree	Density (%)	Reciprocal Ratio (%)	Connected Components
IAPR-TC12	5,000	203,393	81.36	1.627	30.42	1
MIRFlickr	5,000	220,005	88.00	1.760	25.29	1
NUS-WIDE	5,000	200,417	80.17	1.604	36.39	1

To examine whether the constructed residual relation graphs exhibit the structural properties needed for global recovery, Table 4 reports their statistics and Figure 9 provides corresponding visualizations. All three graphs remain sparse, with densities of only 1.604%–1.760%, while each forms a single connected component, showing that locally mined residual relations are successfully integrated into a globally connected structure. The reciprocal ratios range from 25.29% to 36.39%, indicating that a substantial portion of directional residual evidence receives support from the reverse direction. Figure 9 further reveals coherent global connectivity together with structured local neighborhoods. The right-skewed degree distributions, with mean degrees of 80.17–88.00 and lower median degrees, also show heterogeneous connectivity across nodes. These results indicate that ReCapHash constructs sparse yet globally connected residual graphs, providing a suitable structural basis for subsequent global spectral recovery.

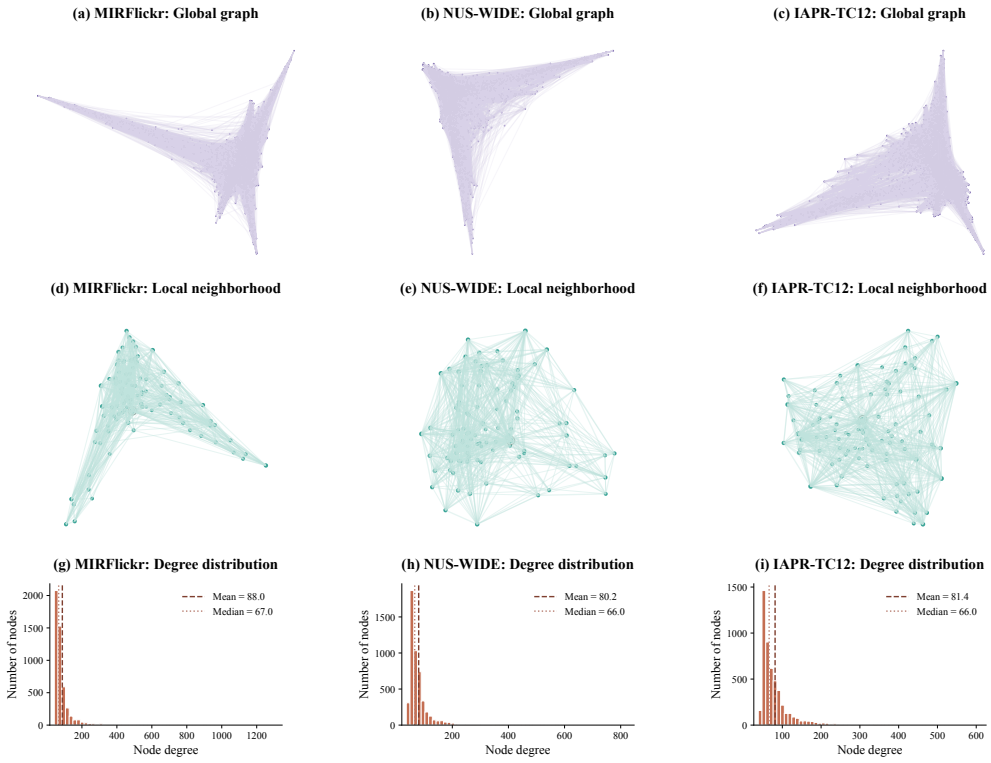


Figure 9: Structural visualization of the residual relation graphs on MIRFlickr, NUS-WIDE, and IAPR-TC12. The three rows show global graphs, local residual neighborhoods, and the corresponding node-degree distributions with mean and median degrees, respectively.

C.5 Diversity-Aware Binary Recovery Analysis

To examine the effect of diversity-aware binary recovery, Table 5 compares a single ITQ restart, selection without diversity regularization, and the full ReCapHash criterion. Compared with a single ITQ restart, selecting from multiple candidate solutions using graph agreement consistently improves both G and retrieval performance, showing the benefit of exploring multiple binary solutions. However, when diversity is ignored ($\lambda_d = 0$), the selected codes achieve the highest graph agreement but also exhibit substantially higher inter-bit correlation. For example, on MIRFlickr, ρ_{bit} increases from 0.3520 to 0.4136, despite the improved graph agreement.

The full ReCapHash criterion balances graph fidelity and bit complementarity. Although its G is slightly lower than that of the $\lambda_d = 0$ variant, it consistently yields the lowest ρ_{bit} and the best retrieval performance across all three datasets. On MIRFlickr, ρ_{bit} decreases from 0.4136 to 0.3168, while I→T and T→I improve from 0.8950/0.8601 to 0.8977/0.8626. Similar trends are observed on NUS-WIDE and IAPR-TC12. These results indicate that maximizing residual graph agreement

Table 5: Effect of diversity-aware binary recovery on graph agreement, bit redundancy, and retrieval performance at $B = 32$. G denotes residual graph agreement, and ρ_{bit} denotes the maximum absolute correlation between recovered bits.

Dataset	Variant	$G \uparrow$	$\rho_{\mathrm{bit}} \downarrow$	I→T↑	T→I↑
MIRFlickr ($K = 12$)	Single ITQ restart	0.7539	0.3520	0.8886	0.8539
	w/o diversity selection ($\lambda_d = 0$)	**0.7586**	0.4136	0.8950	0.8601
	Full ReCapHash	0.7562	**0.3168**	**0.8977**	**0.8626**
NUS-WIDE ($K = 12$)	Single ITQ restart	0.7640	0.2852	0.7890	0.7573
	w/o diversity selection ($\lambda_d = 0$)	**0.7752**	0.3100	0.7929	0.7615
	Full ReCapHash	0.7709	**0.2168**	**0.7953**	**0.7632**
IAPR-TC12 ($K = 7$)	Single ITQ restart	0.8160	0.3440	0.7542	0.7425
	w/o diversity selection ($\lambda_d = 0$)	**0.8263**	0.3608	0.7599	0.7463
	Full ReCapHash	0.8213	**0.2416**	**0.7631**	**0.7497**

alone may favor redundant recovered dimensions, whereas diversity-aware selection produces more complementary residual bits and better retrieval performance.

C.6 Runtime and Scalability Analysis

Table 6: Runtime and scalability of ReCapHash on MIRFlickr at $B = 32$. Runtime is averaged over three independent runs with 2,400 distillation steps at each scale. Runtime is reported in seconds, and CPU/GPU memory in GiB.

N_{train}	Utility	Graph	Spectral+ITQ	Distill	Total	CPU (GiB)	GPU (GiB)	I→T	T→I
1,000	0.25	0.26	0.25	13.63	14.39	1.40	0.08	0.8714	0.8472
2,000	0.29	0.34	0.40	15.25	16.29	1.44	0.09	0.8822	0.8590
5,000	0.49	0.66	0.97	14.88	17.00	1.48	0.13	0.8977	0.8626
10,000	0.86	1.30	1.65	14.41	18.22	1.60	0.26	0.9022	0.8639

To evaluate the computational scalability of ReCapHash, Table 6 reports the runtime, peak memory usage, and retrieval performance as the number of training pairs increases from 1K to 10K. The overall computational overhead grows moderately with the training scale. In particular, increasing N_{train} from 1,000 to 10,000 increases the total runtime from 14.39 s to 18.22 s, while peak CPU and GPU memory increase from 1.40/0.08 GiB to 1.60/0.26 GiB. The Utility, Graph, and Spectral+ITQ stages remain lightweight even at 10K training pairs, requiring only 0.86 s, 1.30 s, and 1.65 s, respectively. Distillation accounts for most of the runtime and remains nearly constant across scales because the number of distillation steps is fixed.

Increasing the training scale also does not degrade retrieval quality. I→T improves from 0.8714 at 1K training pairs to 0.9022 at 10K, while T→I increases from 0.8472 to 0.8639. The standard $N_{\mathrm{train}} = 5{,}000$ setting used in the main experiments achieves 0.8977/0.8626 for I→T/T→I with a total runtime of 17.00 s. These results show that ReCapHash introduces modest computational and memory growth over the evaluated range while maintaining strong retrieval performance as the training set increases.

D Additional Implementation Details

D.1 Dataset Details

We conduct experiments on three widely used public cross-modal retrieval datasets:

- **MIRFlickr-25K** (Huiskes & Lew, 2008) contains 25,000 image–text pairs. We use 20,015 pairs covering 24 semantic categories in our experiments.
- **NUS-WIDE** (Chua et al., 2009) contains 269,648 image–text pairs. We use 186,577 pairs from the 10 most frequent semantic categories.
- **IAPR-TC12** (Escalante et al., 2010) contains 20,000 image–text pairs with 255 semantic labels.

For each dataset, we randomly select 2,000 image–text pairs as the query set and use the remaining pairs as the retrieval set, resulting in 18,015, 184,577, and 18,000 retrieval pairs for MIRFlickr, NUS-WIDE, and IAPR-TC12, respectively. We further sample 5,000 pairs from each retrieval set for training. Following the unsupervised setting, semantic labels are not used during training and are only employed for retrieval evaluation.

D.2 Baseline Details

The baseline methods are summarized as follows:

- **UCCH** (Hu et al., 2022) introduces contrastive learning into unsupervised cross-modal hashing and employs cross-modal ranking learning to alleviate the influence of false negatives.
- **CIRH** (Zhu et al., 2022) jointly preserves cross-modal correlation semantics and modality-specific identity information through correlation-identity reconstruction.
- **UCH-STM** (Tu et al., 2023) mines semantic correlations between words to construct more reliable cross-modal similarities and further refines them during hash learning.
- **HEH** (Zhong et al., 2023) exploits hypergraph structures with robust similarity guidance to capture higher-order relations for unsupervised cross-modal retrieval.
- **CMGCH** (Liang et al., 2024) constructs multimodal knowledge graphs and combines graph contrastive learning with multi-granularity representations to capture fine-grained cross-modal relations.
- **GASKN** (Yu et al., 2024) learns a graph-contrastive semantic anchor space and transfers multimodal knowledge through dynamic knowledge distillation.
- **HuggingHash** (Wang et al., 2024) employs multi-granularity aligned Transformers to enhance multimodal semantic representation and cross-modal alignment.
- **VTM-UCH** (Fan & Cao, 2025) performs vision-guided text mining and community similarity quantization to refine cross-modal similarities and hash-code distributions.
- **USFTH** (Yang et al., 2025) constructs a fused similarity structure and employs Transformer-based interactions to integrate heterogeneous multimodal information.
- **ENHash** (Fu et al., 2026b) introduces error-notebook-guided fine-grained learning to exploit difficult and previously misrepresented cross-modal relations.
- **BRAINHash** (Fu et al., 2026a) adopts a brain-inspired region-aligned interaction framework with memory-based temporal modeling for unsupervised cross-modal hashing.
- **UWMCH** (Yang et al., 2026b) combines masked multimodal fusion with semantic-affinity-guided weighted contrastive learning to improve reliable cross-modal alignment.
- **HINT** (Xiao et al., 2026) constructs a hierarchical cross-modal encoding tree and employs modality mixup to capture multi-level semantic structures and improve modality alignment.

D.3 Implementation Details

For the main experiments, we use frozen CLIP ViT-B/32 features for all baselines and ReCapHash, with modality-specific MLP hash heads generating the initial B-bit codes for the main ReCapHash setting. All experiments are conducted on a server equipped with an Intel(R) Core(TM) i9-14900HX CPU at 2.20 GHz, an NVIDIA GeForce RTX 4070 GPU, and 32 GB of RAM.

D.4 Evaluation Metrics

Following common practice in cross-modal retrieval, we evaluate all methods in two retrieval directions: image-to-text (I→T), where images are used as queries to retrieve texts, and text-to-image (T→I), where texts are used as queries to retrieve images. Retrieval performance is measured using mean Average Precision (mAP) and Top-k Precision (P@k). For mAP, we evaluate the top 50 retrieved instances and report mAP@50. P@k measures the proportion of relevant instances among the top-k retrieved results and is used for the Top-k precision curves.