Multi-view Hashing Classification

Yuhang Lan
College of Computer Science, Sichuan University
Chengdu, China
yhanglan@163.com

Shilin Xu
College of Computer Science, Sichuan University
Chengdu, China
xushilin990@gmail.com

Chao Su
College of Computer Science, Sichuan University
Chengdu, China
suchao.ml@gmail.com

Run Ye
School of Automation Engineering, University of Electronic Science and Technology of China
Chengdu, China
rye@uestc.edu.cn

Dezhong Peng
College of Computer Science, Sichuan University
Tianfu Jincheng Laboratory
Chengdu, China
pengdz@scu.edu.cn

Yuan Sun*
National Key Laboratory of Fundamental Algorithms and Models for Engineering Numerical Simulation, Sichuan University
Chengdu, China
sunyuan_work@163.com

Abstract

Multi-view classification aims to leverage information from multiple views of data to improve prediction performance by learning complementary and consistent representations. Therefore, in recent years, multi-view learning has attracted widespread attention in the community. Despite the success of existing multi-view learning methods, there are still some challenges when dealing with large-scale multi-view data. To address this issue, we propose a novel Multi-view Hashing Classification (MHC) framework to encode large-scale multi-view data as binary codes, thereby enhancing the semantic discrimination. Specifically, we leverage class prompts to generate corresponding textual descriptions for each instance and learn the corresponding anchor hash codes. To achieve intra-class compactness and inter-class separability, we propose Class-prompt Contrastive Learning (CCL) to enforce class-wise aggregation and separation in the Hamming space. To mitigate the cross-view heterogeneity gap, we propose a Supervised Cross-view Contrastive (SCC) module to align view-specific hash codes under label supervision. Finally, we present Boundary-aware Independent Hashing (BIH) that introduces boundary-aware constraints to reduce class boundary ambiguity, thereby improving the discrimination of fusion hash codes. Nevertheless, we observe that anchor hash codes could violate the bit independence assumption, which potentially hinders the optimization direction. To this end, we adopt a Bit-level Calibration Mechanism (BCM) to filter out redundant bits, thereby restoring bit independence. Extensive experiments conducted on ten benchmark datasets demonstrate the superiority of the proposed MHC in terms of both classification accuracy and inference efficiency. The code is released at https://github.com/Yuhang-lan04/MHC.

*Corresponding author

MM '25, Dublin, Ireland.

ACM ISBN 979-8-4007-2035-2/2025/10
https://doi.org/10.1145/3746027.3755692

CCS Concepts

• **Computing methodologies** → **Supervised learning by classification**.

Keywords

Multi-view Classification, Hashing Learning, Boundary Preserving

ACM Reference Format:
Yuhang Lan, Shilin Xu, Chao Su, Run Ye, Dezhong Peng, and Yuan Sun. 2025. Multi-view Hashing Classification. In *Proceedings of the 33rd ACM International Conference on Multimedia (MM '25), October 27–31, 2025, Dublin, Ireland*. ACM, New York, NY, USA, 10 pages. https://doi.org/10.1145/3746027.3755692

1 Introduction

In real-world applications, multi-view data refers to data instances that are described from multiple distinct perspectives or modalities [35]. Each view provides some unique information that reflects the intrinsic characteristics of the object. For instance, an image can be described by RGB, depth, and texture features, while a medical case might integrate MRI scans, clinical reports, and genomic profiles. These different views inherently capture complementary information, thereby endowing a more comprehensive understanding of the underlying objects. Therefore, in the era of big data, multi-view data has become increasingly prevalent. Multi-view classification (MVC) [33, 34], aiming to improve decision-making by effectively leveraging information from multiple views, has become a hot research topic. Unlike single-view learning, which may suffer from incomplete or biased information, multi-view methods [36, 37] could exploit the consistency and complementarity among views to enhance the accuracy and robustness of the learning model. For example, in social media analysis, combining text, image, and user interaction views can significantly improve sentiment classification performance. Therefore, with the explosive growth of multi-view data, MVC has garnered great attention in various application fields, such as autonomous driving [2], precision medicine [20, 23], and cross-modal retrieval [14, 16, 31].

Existing MVC methods could be broadly categorized into two primary types, i.e., end-to-end classification model [8, 18] and representation learning model [30]. The first line of methods [3, 32, 40]

usually employs deep neural networks to map from input data to the final category label directly. In brief, the network outputs the category probability distribution or specific category index. The second line of methods [6, 17] first adopts deep neural networks to learn discriminative representations from input data automatically, and then uses an independent classifier (such as MLP, SVM, k-NN) for classification. Since representation learning methods can learn more universal and discriminative representations, they are more adaptable to different downstream tasks, such as classification, clustering, and retrieval. Thus, multi-view representation learning has received extensive attention in recent years.

Although existing multi-view representation learning methods [6, 17, 30] have demonstrated promising performance, high computational cost remains a key challenge in large-scale multi-view learning scenarios. Fortunately, hashing learning[19] provides an effective solution due to its excellent computational efficiency. Some multi-view hashing learning methods [41] have been proposed, which aim to encode multi-view data into short-length hash codes while preserving the original semantic similarity in Hamming space. However, all of these methods are specifically designed for clustering tasks [15, 28, 29], which are difficult to be directly used for multi-view classification. Recently, a few hashing methods have been proposed for classification tasks, such as unimodal classification [21] or image set classification [24, 24]. Nevertheless, it is still a less-touched problem about multi-view hashing classification. The core challenge lies in how to learn compact hash codes to alleviate the multi-view heterogeneous gap and improve the semantic discrimination of hash codes.

To address these challenges, we propose a multi-view hashing classification paradigm (MHC) for large-scale multi-view data, which could enhance the semantic discrimination of hash codes and improve reasoning efficiency. As shown in Fig.1, our MHC consists of three core modules, i.e., Class-prompt Contrastive Learning (CCL), Supervised Cross-view Contrastive (SCC), and Boundary-aware Independent Hashing (BIH). Specifically, we first obtain the text description of each instance through a set of predefined class prompts and further learn corresponding text hash codes. These text hash codes encapsulate the semantic essence of each class and serve as clear reference points, guiding multi-view data toward their corresponding classes in the feature space. Consequently, we regard these text hash codes as anchor hash codes in the subsequent process. In addition, we generate the view-specific hash codes for each view and fuse them to get the fusion hash code. To improve inter-class discrimination, we propose the CCL module to enforce tight aggregation of text hash codes from the same class in the Hamming space while ensuring clear separation between ones of different classes. To mitigate heterogeneous differences and explore cross-view consistency, under the guidance of class labels, we present the SCC module to minimize the similarity between cross-view hash codes from the same class, while maximizing the similarity between those of different classes. To enhance the discrimination of the fusion hash codes, we propose the BIH module to maintain clear decision boundaries between dissimilar fusion and anchor hash codes, thereby preventing category margin ambiguity. Moreover, the generated anchor hash codes could ignore or violate the bit independence assumption, thereby resulting in an equal probability of +1 and −1 appearing in a particular bit. This could avoid fusion hash codes towards the wrong optimization direction when pulling them close to anchor hash codes. To this end, we further propose a Bit-level Calibration Mechanism(BCM) to filter these redundant bits, thereby endowing the bit independence character. In summary, the major contributions of this paper are as follows:

- We propose a new Multi-view Hashing Classification paradigm (MHC) that transforms multi-view data into a more compact representation, thereby improving the semantic discrimination and achieving efficient classification. To the best of our knowledge, this is the first work to bulid the hashing model to handle large-scale multi-view classification tasks.
- We propose boundary-aware independent hashing that keeps clear decision boundaries to mitigate category margin ambiguity while filtering out irrelevant bits to prevent falling into the wrong optimization direction.
- We perform extensive experiments compared with 12 state-of-the-art baselines on ten publicly available multi-view datasets. The results demonstrate that our MHC outperforms these competitors in terms of accuracy and efficiency.

2 Related Work

2.1 Multi-view Classification

Multi-view classification aims to leverage complementary information from multiple heterogeneous views or modalities to enhance predictive performance. Existing approaches can be broadly categorized into two types: end-to-end learning models and representation learning models. End-to-end models directly predict class labels by processing raw multi-view inputs within a unified architecture. For example, TMC [8] and ETMC [9] enhance the reliability and robustness of classification by dynamically integrating different views at the evidence level. However, these models overlook the issue of view incompleteness, which frequently arises in real-world scenarios. To address this issue, UIMC [32] models the uncertainty of missing views by repeatedly constructing distributions and samples, adapting them based on sampling quality. Nevertheless, UIMC assumes strictly aligned views, whereas in practical applications, multi-view data may contain conflicting or inconsistent instances. To handle such cases, ECML [33] introduces a conflict opinion aggregation strategy that provides more reliable outcomes for conflicting inputs. In contrast, representation learning models aim to learn view-invariant and discriminative features. By decoupling feature learning from the decision-making process, these models offer greater generalization capability. For instance, DCCAE [30] combines deep CCA [1] with autoencoders to learn compact and informative representations, further improving the capacity to model complex multi-view dependencies. However, DCCAE overlooks the dynamic noise present in multi-view data. To mitigate this, DUA-Nets [6] integrate intrinsic information from multiple views to obtain noise-free representations, guided by data uncertainty estimated from a generative perspective. Furthermore, to jointly address cross-view consistency and missing view recovery, DCP [17] proposes an information-theoretic framework that maximizes mutual information across views via contrastive learning, while minimizing conditional entropy through dual prediction to achieve both information consistency and data recoverability. Although

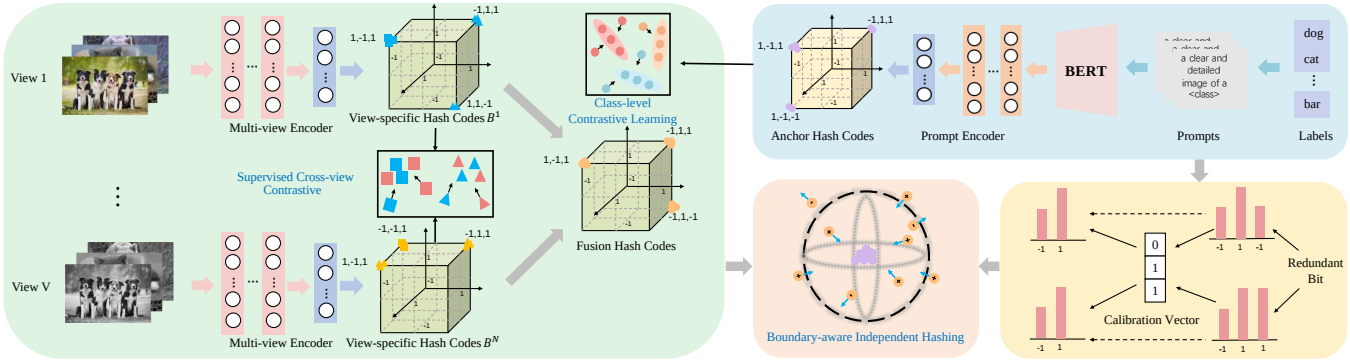


Figure 1: The framework of our MHC. Specifically, we first use predefined prompt templates to extend labels, thereby generating anchor hash codes. Then, we propose Class-level Contrastive Learning (CCL) to ensure intra-class compactness and inter-class separation of specific-view hash codes. Then, we propose Supervised Cross-view Contrastive (SCC) to aggregate fusion hash codes from the same class. Finally, we propose Boundary-aware Independent Hashing (BIH) to enforce clear decision boundaries and remove redundant bits of anchor hash codes, thereby enhancing the discrimination of fusion hash codes.

their effectiveness, current representation learning approaches often suffer from high inference latency, especially when applied to large-scale multi-view datasets. This issue primarily stems from their reliance on real-valued feature representations, which introduce substantial computational overhead during the prediction phase and limit their scalability and practicality in time-sensitive scenarios. In contrast, our approach is designed to generate compact binary representations, enabling efficient inference while maintaining strong classification performance. This makes it more suitable for large-scale multi-view learning scenarios.

2.2 Supervised Cross-modal Hashing

Hashing learning, renowned for its compact storage and rapid retrieval, has become indispensable for cross-modal retrieval. Traditional shallow hashing methods, which rely on manually crafted feature descriptors, often struggle to capture complex semantic relationships within data, leading to poor performance[39, 42]. To address these limitations, recent cross-modal hashing frameworks integrate feature learning and hash code generation within an end-to-end architecture, significantly improving retrieval accuracy[10, 22, 25]. For instance, the SSAH[13] framework introduces adversarial learning into cross-modal hashing by employing two adversarial networks to maximize the semantic correlation and consistency between modalities. In addition, researchers have proposed triplet-based methods, such as TDH[4] and AGAH[7], which combine triplet loss with asymmetric loss functions to capture higher-order relationships, thereby improving retrieval performance through global dataset interactions. However, these methods still face challenges related to binary optimization and quantization errors. To mitigate these issues, the DCHMT[26] method introduces a multimodal transformer to capture positional information in images and employs differentiable optimization strategies to avoid the difficulties associated with traditional discrete binary optimization. Despite these advances, most methods rely heavily on data-point similarity and label supervision to guide the learning of the hashing model. However, such similarity measures only partially reflect label semantics, and often fail to capture fine-grained semantic details, limiting overall retrieval performance. To overcome this, the DCPH[27] framework redefines the learning objective by introducing category-level proxies, generating compact hash codes without explicitly computing pairwise similarities. Nevertheless, this approach may still lack the capacity to model the full spectrum of semantic relationships in heterogeneous data, resulting in coarse-grained representations. To address this, DSPH[11] framework extends the proxy mechanism, embedding multi-label data into a unified discrete space and capturing fine-grained semantic relations among original samples. Inspired by the efficiency and flexibility of hashing learning, we extend its application beyond cross-modal retrieval to the field of multi-view classification. Further, we propose a novel hashing framework for multi-view classification, thereby effectively enhancing both efficiency and discriminative capability in large-scale multi-view data scenarios.

3 Method

3.1 Notation

Multi-view classification (MVC) aims to develop a hashing model to effectively integrate the consistent and complementary information from all available views, thereby accurately predicting the category of an unseen instance. In this paper, the multi-view data is denoted as $O = \{\{x_i^v\}_{v=1}^V, y_i\}_{i=1}^N$, where N, V, x_i^v represent the number of instances, the number of views, and the $i-th$ sample from the v-th view, respectively. $y_i \in \mathbb{R}^C$ represents the corresponding one-hot label of the i-th instance, where C is the number of categories. The feature dimension of each sample from the v-th view is denoted as d^v. The goal of our MHC is to map multi-view data into L-bit binary codes $h_i^v \in \{-1, 1\}^L$ in Hamming space, where h_i^v represents hash codes of the i-th sample from the v-th view.

During the optimization phase, since binary codes are non-differentiable, we employ a continuous relaxation scheme that derives a binary-like vector using the *tanh* function. The formula could be written as

$$b_i^v = tanh\left(H^v\left(x_i^v\right)\right) \in (-1, 1)^L, \tag{1}$$

where H^v represents the hash function of each view. For the testing phase, we adopt the *sign* function [12] to convert the continuous binary-like vector b_i^v into binary hash codes h_i^v, i.e.,

$$h_i^v = sign\left(b_i^v\right). \tag{2}$$

3.2 Class-prompt Contrastive Learning

To enhance the semantic richness and diversity of label representations in multi-view data, we adopt a set of text prompts to generate textual category descriptions for each label. This strategy can ensure that the same labels exhibit different expressions, thereby establishing the semantic correlations between the classes. Specifically, we first construct 10 predefined prompt templates (see Appendix), which describe the characteristics or attributes of a label from a unique perspective. Then, we obtain the textual descriptions for each instance by randomly selecting a prompt from a predefined set and concatenating it with the label. For instance, a label such as 'dog' could be expanded into semantically enriched descriptions like 'A clear and detailed image of a dog', 'A high-quality picture of a dog', or 'A crisp and vibrant photo of a dog', depending on the chosen prompt. Thus, the prompt template could be written as

$$p = \text{ A clear and detailed image of a <class>}, \tag{3}$$

where <class> is the category name. Afterward, we employ the BERT model [5] to convert the generated category description text into the corresponding category text embedding representation $\mathcal{F}_i^T$. Further, we learn the corresponding hash codes by hash projection H^T, i.e.,

$$m_i = tanh\left(H^T\left(\mathcal{F}_i^T\right)\right). \tag{4}$$

Since they contain class information, we regard them as the anchor hash codes.

To mine intra-class consistency, we present Class-level Contrastive Learning (CCL), which maximizes intra-class similarity while minimizing inter-class similarity. Specifically, we first calculate the cosine similarity between any two representations, i.e.,

$$\mathrm{S}\left(m_i, m_j\right) = \frac{< m_i, m_j >}{|m_i||m_j|}, \tag{5}$$

where $< m_i, m_j >$ denotes the inner product. Then, we adopt the CCL loss as follows

$$\mathcal{L}_{CCL} = -\frac{1}{N}\sum_{i=1}^{N}\log\frac{\sum_{j=1}^{N} I_{ij}\cdot\exp\left(\mathrm{S}\left(m_i, m_j\right)/\tau\right)}{\sum_{k=1}^{N}\exp\left(\mathrm{S}\left(m_i, m_k\right)/\tau\right)}. \tag{6}$$

where τ is a temperature parameter that controls the sharpness of the similarity distribution. I_{ij} is the indicator that can be defined as

$$I_{ij} = \begin{cases} 1, & y_i = y_j \\ 0, & y_i \neq y_j \end{cases} \tag{7}$$

In general, CCL explicitly optimizes the distribution and alignment of label descriptions in Hamming space. This promotes that anchor hash codes from the same class are tightly clustered to enhance intra-class consistency, while ones from different classes are well-separated to improve inter-class discrimination.

3.3 Supervised Cross-view Contrastive

To effectively explore the cross-view consistency information in multi-view data, we propose supervised cross-view contrastive(SCC) to learn discriminative representations, which maximizes the similarity of cross-view samples of the same class and minimizes the similarity of cross-view samples of different classes under the guidance of labels. To be specific, we first compute the similarity of cross-view positive pairs for each sample as follows

$$Q_{pos}(v,i) = \sum_{u=1,u\neq v}^{V}\sum_{j=1}^{N} I_{ij}\cdot\exp\left(\mathrm{S}\left(b_i^v, b_j^u\right)/\tau\right), \tag{8}$$

Afterward, the overall similarity of cross-view pairs could be represented as

$$\mathrm{Q}_{\text{all}}(v,i) = \sum_{z=1}^{V}\sum_{j=1}^{N}\exp\left(\mathrm{S}\left(b_i^v, b_j^z\right)/\tau\right). \tag{9}$$

Thus, the SCC loss could be formulated as

$$\mathcal{L}_{SCC} = -\frac{1}{\mathrm{V}\cdot N}\sum_{v=1}^{V}\sum_{i=1}^{N}\left[\log\left(\frac{Q_{pos}(v,i)}{Q_{\text{all}}(v,i)}\right)\right]. \tag{10}$$

Afterward, we take the mean for all view-specific representations of each sample as its fusion representation. Formally, fusion hash codes e_i could be computed as

$$\mathrm{e}_i = \frac{1}{V}\sum_{j=1}^{V} b_i^j \tag{11}$$

In general, SCC can make the cross-view representations of the same category closer and the cross-view representations of different categories more separated.

3.4 Boundary-aware Independent Hashing

For multi-view classification tasks, achieving clear class separability between classes in the feature space remains a key challenge. When learning representations from multiple views, the model could produce blurred decision boundaries between different classes. In brief, some semantically similar instances with different classes may overlap, thereby leading to a decline in classification accuracy.

To address the blurred decision boundaries, we design a Boundary-aware Hashing (BH) strategy to enforce a clear semantic margin between different classes. Specifically, this strategy promotes that the fusion hash codes of each instance are tightly aggregated with the anchor hash codes from the same class while maintaining a certain distance from different classes. The BH loss is expressed as

$$\begin{aligned}\mathcal{L}_{BH} = \frac{1}{N(N-1)}\sum_{i=1}^{N}\sum_{j=1}^{N}\Bigg[I_{ij}\cdot\left(\frac{1-\mathrm{S}\left(e_i, m_j\right)}{2}\right)^2 \\ +\left(1-I_{ij}\right)\cdot\mathrm{MAX}\left(0, \xi-\frac{1-\mathrm{S}\left(e_i, m_j\right)}{2}\right)^2\Bigg].\end{aligned} \tag{12}$$

Clearly, the BH loss minimizes the distance for positive pairs to promote intra-class compactness and enforces the boundary value for negative pairs to encourage inter-class separability. The parameter ξ controls the degree of separation between representations of different classes. Specifically, it enforces a boundary value ξ about the cosine similarity for negative pairs, thereby ensuring that the

fusion hash codes e_i are sufficiently distinct from the anchor hash codes from other classes.

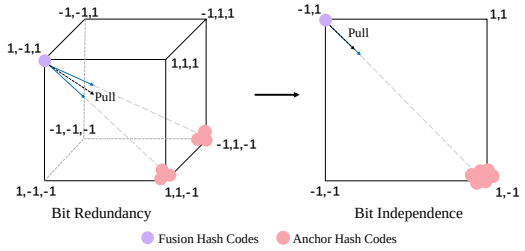


Figure 2: The optimization process of bit-level calibration.

However, BH greatly ignores the distribution character of anchor hash codes, i.e., bit independence. Once the probabilities of 1 and -1 occurring in specific bits are equal, these redundant bits contain no meaningful semantic information. As shown in Fig.2, this could lead to the fusion hash codes trend in the wrong optimization direction during the process of aggregation to anchor hash codes from the same class. To prevent the wrong optimization direction caused by redundant bits, we propose a bit-level calibration mechanism (BCM). For the i-th sample, we compute the expectation of each bit in the anchor hash codes that belong to the same class. Specifically, we first conduct a bit-wise summation operation for each bit through the following formula,

$$\mu_i = \frac{1}{|\Omega(i)|} \sum_{j \in \Omega(i)} \text{sign}\left(m_j\right) \tag{13}$$

where $\Omega(i)$ represents the sample set that shares the same label as the i-th sample, which are considered positive pairs.

Then, we construct a bit-level calibration vector for each bit, i.e.,

$$\delta_i = \begin{cases} 1, & |\mu_i| \geq \varphi \\ 0, & |\mu_i| < \varphi \end{cases} \tag{14}$$

where φ is a threshold. If the absolute expectation of anchor hash codes from the same class falls below the predefined threshold φ in a certain bit, the corresponding bit is regarded as redundant. To filter out the redundant bit, we set the calibration vector of the bit to 0, otherwise, we set it to 1. Finally, we integrate BH and BCM to obtain our BIH loss as follows

$$\begin{aligned} \mathcal{L}_{BIH} = \frac{1}{N(N-1)} \sum_{i=1}^{N} \sum_{j=1}^{N} \Bigg[I_{ij} \cdot \left(\frac{1 - \text{sim}\left(e_i, \delta_j \cdot m_j\right)}{2} \right)^2 \\ + \left(1 - I_{ij}\right) \cdot \text{MAX}\left(0, \xi - \frac{1 - \text{sim}\left(e_i, \delta_j \cdot m_j\right)}{2} \right)^2 \Bigg] \end{aligned} \tag{15}$$

In general, BIH could effectively preserve clear decision boundaries between different categories while filtering out redundant bits in the anchor hash codes, thereby enhancing the discriminative ability.

3.5 Overall Loss

By combining the above losses, we can formulate the overall loss of the proposed MHC as follows

$$\mathcal{L}_{MHC} = \mathcal{L}_{CCL} + \alpha \mathcal{L}_{SCC} + \beta \mathcal{L}_{BIH} \tag{16}$$

where α and β are the balance parameters. The pseudo-code details of the training process are provided in the supplementary material.

3.6 Multi-view Classification

To achieve multi-view classification, we first generate hash codes of the test multi-view data. Then, we calculate Hamming distances between the training and testing data using the following formula

$$D_i = \frac{1}{2}\left(L - < u_t, u_i >\right), \quad i = 1, \ldots, N, \tag{17}$$

where u_t and $u_i = sign(e_i)$ are hash codes of the i-th testing sample and training sample, respectively.

To predict the label of the testing sample, we retrieve the training sample with the minimum Hamming distance D_i. The index i^* of the selected training sample is determined by

$$i^* = \arg\min_i D_i. \tag{18}$$

Finally, the label $\hat{y}_{\text{test}}$ of each test sample could be predicted as the corresponding label y_{i^*} of the most similar training sample, i.e.,

$$\hat{y}_{\text{test}} = y_{i^*}. \tag{19}$$

4 Experiments

4.1 Datasets

We comprehensively evaluate the classification performance of the proposed MHC on ten widely used datasets: MSRC, Leaves, UCI, HW, LandUse, Scene, ALOI-100, Animal, YoutubeFace, and ALOI-1000. Detailed statistics are provided in the supplementary material.

4.2 Compared Methods

To validate the effectiveness of the proposed MHC, we compare it with 12 state-of-the-art MVC methods, which contains end-to-end learning models (i.e., TMC[8], TMDLO[18], ETMC[9], UIMC[32], QMF[40], ECML [33], TMNR[34], and PDF[3]) and representation learning models (i.e., DCCAE[30], CPM-Nets[38] DUA-Nets[6], and DCP[17]). For a fair comparison, we use the recommended parameters of the original paper for all compared methods. Each experiment is repeated five times, and we report the average classification accuracy and standard deviation to ensure statistical reliability. 'O/M' represents out of memory. The best and second-best results are highlighted in **boldface** and underlined, respectively.

4.3 Implementation Details

All of our experiments are implemented in PyTorch and conducted on an NVIDIA GeForce RTX 3090. During the training stage, we employ fully connected neural networks to process both the multi-view data and the label description texts. The multi-view network, which encodes each view of the input data, consists of three stacked hidden layers, with the following dimensions: D_v-8192-8192-8192-L, where D_v denotes the input feature dimension of a view, and L is the length of the generated hash codes. The text network, responsible for encoding the label descriptions, comprises two hidden layers with dimensions: D_t-8192-8192-L, where D_t refers to the BERT-based input embedding dimension. Except for the final layer, each layer is followed by a Rectified Linear Unit (ReLU) activation function. We use the Adam optimizer to train all models. For all datasets,

Table 1: Classification accuracy (%) of our MHC and twelve compared methods on the first five datasets.

Method	References	MSRC	Leaves	UCI	HW	LandUse	Mean
TMC	ICLR'20	93.81±2.43	52.44±2.94	96.55±0.46	96.00±0.81	49.62±3.26	77.68±1.98
TMDLO	AAAI'22	80.10±6.02	90.81±1.55	94.55±0.70	94.30±1.02	26.52±0.94	77.26±2.05
ETMC	TPAMI'22	83.33±7.22	79.75±2.56	96.45±0.40	95.95±0.29	45.00±3.04	80.10±2.70
UIMC	CVPR'23	97.62±0.00	97.06±0.47	97.40±0.34	98.20±0.37	68.00±1.58	91.66±0.49
QMF	ICML'23	91.90±2.43	84.88±2.30	96.45±0.83	96.55±0.76	38.52±1.85	81.66±1.63
ECML	AAAI'24	84.29±7.00	73.19±1.46	84.55±1.43	91.40±0.78	37.95±1.94	74.28±2.52
TMNR	IJCAI'24	90.95±3.16	68.06±5.85	96.90±0.65	97.65±0.49	41.71±2.46	79.05±2.52
PDF	ICML'24	96.19±2.43	97.88±0.54	97.75±0.57	98.40±0.25	44.19±1.57	86.88±1.07
DCCAE	ICML'15	32.38±6.32	61.88±3.97	91.35±1.91	76.90±2.82	32.62±1.74	59.03±3.35
CPM-Nets	NIPS'19	14.29±1.51	23.75±1.44	91.95±0.40	10.00±0.00	22.57±0.49	32.51±0.77
DUA-Nets	AAAI'21	80.00±3.23	88.00±1.75	94.95±0.93	96.05±0.94	47.05±1.23	81.21±1.62
DCP	TPAMI'23	93.81±4.90	90.62±6.21	98.30±0.37	98.15±0.37	71.38±2.29	90.45±2.82
MHC	Our	**98.57±1.17**	**98.50±0.07**	**98.98±0.37**	**99.00±0.69**	**78.81±1.86**	**94.77±0.83**

Table 2: Classification accuracy (%) of our MHC and twelve compared methods on the last five datasets.

Method	References	Scene	ALOI-100	Animal	YoutubeFace	ALOI-1000	Mean
TMC	ICLR'20	72.11±1.14	85.42±0.94	38.66±0.54	45.23±0.45	0.06±0.01	48.30±0.62
TMDLO	AAAI'22	42.76±3.49	65.15±1.41	54.10±0.67	29.77±0.60	0.89±0.01	38.53±1.24
ETMC	TPAMI'22	70.30±1.31	43.53±1.77	58.86±0.73	76.48±0.23	65.09±0.64	62.85±0.92
UIMC	CVPR'23	75.25±0.33	97.88±0.09	50.72±0.41	78.06±0.27	O/M	75.48±0.28
QMF	ICML'23	57.46±1.01	69.09±0.61	40.28±0.77	56.73±0.20	61.80±0.28	57.07±0.57
ECML	AAAI'24	59.49±1.39	63.46±0.99	48.82+0.39	46.83±1.11	2.47±0.18	44.21±0.81
TMNR	IJCAI'24	63.84±1.40	61.91±1.31	59.99±1.35	O/M	O/M	61.91±1.35
PDF	ICML'24	70.01±0.71	96.83±0.25	63.10±0.47	85.80±0.22	89.18±0.16	80.98±0.36
DCCAE	ICML'15	49.50±1.54	56.71±1.81	31.86±0.64	26.42±0.15	0.15±0.03	32.93±0.83
CPM-Nets	NIPS'19	35.09±0.50	1.07±0.18	7.47±0.54	O/M	O/M	14.54±0.24
DUA-Nets	AAAI'21	52.91±1.48	97.22±0.03	31.58±0.69	59.02±0.51	76.49±0.80	63.44±0.70
DCP	TPAMI'23	74.52±2.41	96.56±0.65	37.91±1.13	74.55±5.54	12.92±4.08	59.29±2.76
MHC	Our	**80.71±0.60**	**99.51±0.12**	**63.27±0.67**	**87.40±0.15**	**92.52±0.17**	**84.68±0.21**

we train for 100 epochs with a batch size of 128 and a learning rate of 3e-5, with the hash code dimension fixed to 128. Based on the bit-level calibration threshold analysis, the threshold(φ) is set to 0.05. Based on the parameter analysis of the balancing parameters α and β, α and β are set to 0.0001 and 1, respectively. Additionally, the temperature parameter(τ) and boundary value(ξ) are empirically set to 0.2 and 0.05. All datasets are randomly split into training and testing sets using an 8:2 ratio.

4.4 Experimental Results and Analysis

We evaluate the proposed MHC against 12 state-of-the-art MVC methods. The experimental results on 10 multi-view datasets are reported in Tab.1 and Tab.2. From these results, we draw the following key observations:

- Compared with all state-of-the-art MVC methods, our MHC achieves the best classification performance. This demonstrates the effectiveness and superiority of our approach in learning discriminative representations for multi-view data and its strong generalization ability across diverse datasets.
- Our MHC achieves significantly higher average performance across all datasets compared to the baseline methods, highlighting the effectiveness of our class-prompt contrastive learning and supervised cross-view contrastive. Additionally, MHC exhibits low variance on large-scale datasets, indicating excellent scalability and robustness in handling high-dimensional and complex multi-view inputs.
- On large-scale datasets with many categories (e.g., ALOI-1000), some baseline methods experience a noticeable decline in performance, whereas MHC consistently maintains high classification accuracy. This advantage can be attributed to our boundary-aware independent hashing, which removes redundant bits from the anchor hash codes while enforcing semantic margins between different classes to enhance inter-class separability.

4.5 Time Cost

To evaluate the inference efficiency of our MHC, we record the time cost of MHC and four representation learning models on five large-scale datasets. Inference time is defined as the duration from feeding the test data into the model to obtaining the final prediction results. The experimental results are presented in Tab.3. These results demonstrate a clear advantage of our MHC in inference efficiency, particularly for large-scale data. This advantage stems from the fundamental difference in similarity computation. While representation learning methods rely on real-valued representations and costly floating-point operations, whose complexity scales with data size and dimensionality, MHC leverages compact binary hash codes and efficient Hamming distance calculations, substantially reducing computational overhead and enhancing scalability. From Tab.1 and Tab.2, we can observe that MHC achieves significantly faster inference without compromising classification accuracy. This highlights the capability of our MHC to effectively balance predictive performance and computational efficiency, particularly when handling large-scale multi-view datasets.

Table 3: Inference time (seconds) of our MHC compared with several representation learning MVC methods on five large-scale datasets.

Method	Scene	ALOI-100	Animal	YoutubeFace	ALOI-1000
DCCAE	0.087	6.44	0.26	7.50	2824.99
DUA-Nets	0.87	2.70	3.08	31.53	23.03
DCP	0.035	0.28	0.18	12.49	19.29
CPM-Nets	1.99	2.84	27.43	O/M	O/M
Our MHC	**0.027**	**0.08**	**0.13**	**1.04**	**0.83**
Improve	Δ1.30	Δ3.50	Δ1.38	Δ7.21	Δ34.98

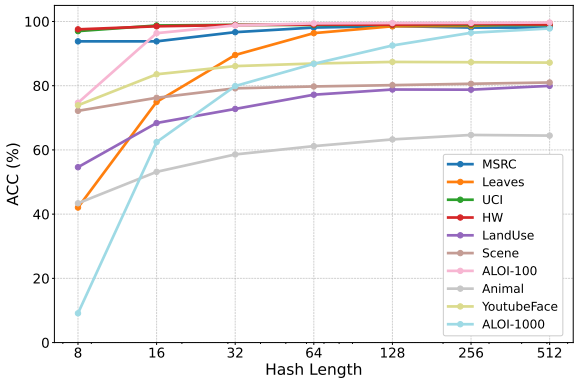


Figure 3: Classification accuracy of our MCH method with different bit lengths on all multi-view datasets.

4.6 Bit Analysis

To evaluate the impact of different bit lengths on the performance, we perform the bit analysis experiment on all multi-view datasets. To be specific, we vary hash lengths to {8, 16, 32, 64, 128, 256, 512} to obtain the classification performance, as shown in Fig.3. When the hash length is too short, there is an increased likelihood of hash collisions, whereby samples from different classes may be mapped to the same binary code. This reduces the model's discriminative capacity and ultimately decreases classification accuracy. As the bit length increases, the classification performance also increases. This could be because long hash codes can contain richer semantic information. When the hash length exceeds 128 bits, the incremental gains in accuracy diminish as the length increases. This indicates that 128-bit hash codes are already sufficient to capture the important discriminative information of multi-view data. Thus, to reduce the unnecessary storage and computational costs, we set the bit length to 128 in our experiments, thereby achieving a balance between performance and efficiency.

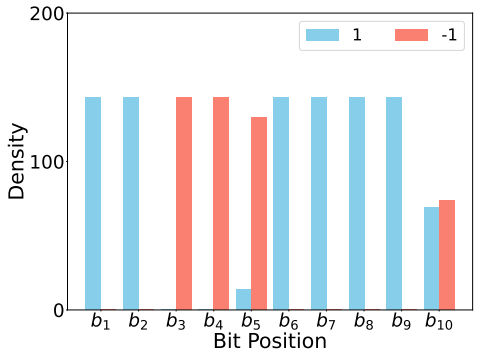


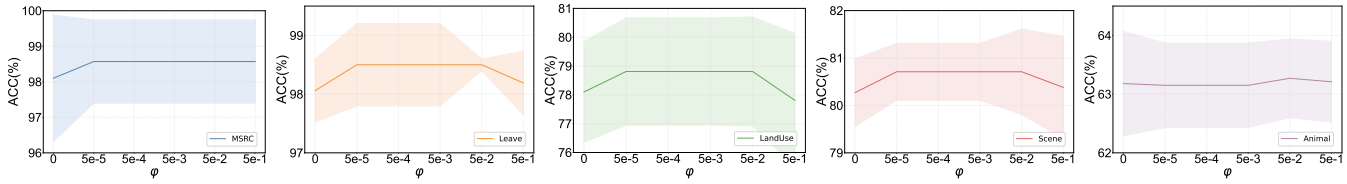
Figure 4: The density versus different bits on the HW dataset.

4.7 Bit-level Calibration Analysis

To show the proposed BCM, we conduct the bit-level calibration analysis experiment on the HW dataset. To be specific, we first select the corresponding text hash codes of 159 text descriptions from a single class. Then, we randomly choose 10 bits for all text hash codes and plot histograms of the frequency distribution, as shown in Fig.4. According to these results, for original hash codes, the frequency distribution of -1 and 1 differs significantly across most bits. However, for a few bits, the frequency distribution of -1 and 1 is relatively balanced or even close to uniform(e.g. b_{10}). According to the bit independence assumption, if the values 1 and -1 occur with equal probability in a particular bit, this bit could fail to convey meaningful discriminative semantic information, essentially representing random noise. Moreover, this bit could misdirect the optimization during the learning process. Thus, we adopt bit-wise calibration to remove several redundant bits that do not provide useful information. Specifically, b_{10} could be removed based on the threshold determined during the calibration process.

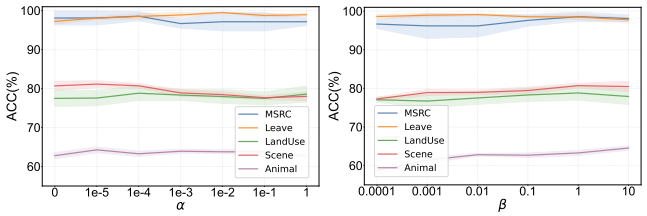
4.8 Threshold Analysis

To investigate the effect of our bit-level calibration threshold φ on classification performance, we conduct the threshold analysis experiment on five datasets (i.e., MSRC, Leaves, LandUse, Scene,



(a) MSRC (b) Leaves (c) LandUse (d) Scene (e) Animal

Figure 5: Threshold analysis (i.e., φ) on five multi-view datasets.



(a) ACC vs. α (b) ACC vs. β

Figure 6: Parameter analysis on five multi-view datasets.

and Animal). Specifically, we set the φ value to 0, 5e-5, $\cdots$, 5e-1, and plot the performance curves, as shown in Fig.5. When the threshold is set to zero or a low value, redundant bits are not effectively eliminated, thereby disrupting the model's optimization and leading to the degradation of classification performance. As the threshold increases, classification accuracy improves and tends to stabilize. This indicates that the redundant bits interfering with optimization could be effectively removed. However, if the threshold is increased excessively, classification accuracy decreases and variance increases. This shows that overly high thresholds mistakenly classify discriminative bits as redundant and remove them. In general, the great threshold could result in the loss of valuable semantic information, thereby weakening the discriminative capacity of hash codes and degrading overall model performance.

4.9 Parameter Analysis

To study the impact of the parameters in our CMH, we conduct a series of parameter analysis experiments for the parameters α and β. To be specific, we vary the values of α and β to $\{1e-5, 1e-4, \cdots, 1\}$ and $\{1e-4, 1e-3, \cdots, 10\}$, respectively. As shown in Fig.6, we plot classification performance curves for different values α and β on five datasets (i.e., MSRC, Leaves, LandUse, Scene, and Animal). From the experimental results, the accuracy initially increases and then decreases as α increases, with optimal performance observed between $1e-5$ and $1e-4$. Similarly, when β is between 1 and 10, we could obtain the optimal classification performance.

4.10 Ablation Study

To validate the effectiveness of each component of our method, we conducted an ablation study on five datasets. The experimental results are presented in Tab.4. When we avoid BCM to ignore redundant bits, the model could be optimized in the wrong direction, thereby resulting in decreased classification accuracy. When semantic boundaries are not explicitly enforced between different categories (i.e., removing the $\mathcal{L}_{BIH}$), the model struggles to maintain clear class separability, thereby leading to increased confusion between similar categories and a performance drop. Once $\mathcal{L}_{CCL}$ and $\mathcal{L}_{SCC}$ are removed, the model's performance significantly decreases, which indicates that $\mathcal{L}_{CCL}$ and $\mathcal{L}_{SCC}$ are crucial for promoting intra-class consistency and inter-class separability. In summary, these experiments demonstrate that all components of our MHC framework are indispensable.

Table 4: Ablation study (%) on five multi-view datasets.

$\mathcal{L}_{CCL}$	$\mathcal{L}_{SCC}$	$\mathcal{L}_{BIH}$	BCM	MSRC	Leaves	LandUse	Scene	Animal
✓	✓	✓	X	<u>98.10</u>	98.06	<u>78.10</u>	80.27	<u>63.18</u>
✓	✓	X	X	96.67	**98.62**	76.95	77.97	57.45
X	X	✓	✓	54.76	80.56	70.38	76.30	40.33
X	X	✓	X	57.14	84.87	73.52	77.99	47.33
✓	✓	✓	✓	**98.57**	<u>98.50</u>	**78.81**	**80.71**	**63.27**

5 Conclusion

In this paper, to deal with large-scale scenarios, we propose a novel Multi-view Hashing Classification (MHC) framework that leverages hashing learning to enhance both computational efficiency and classification. Specifically, we first utilize class prompts to generate the anchor hash codes. Then, we propose Class-prompt Contrastive Learning (CCL) to explore the class consistency. Afterward, we propose Supervised Cross-view Contrastive (SCC) to effectively mitigate cross-view heterogeneity and improve intra-class compactness. Finally, we present Boundary-aware Independent Hashing (BIH) that establishes clear decision boundaries between classes to enhance the semantic discrimination and filters out redundant bits in anchor hash codes to prevent the wrong optimization direction. Extensive experiments on ten widely used datasets demonstrate that MHC outperforms 12 state-of-the-art MVC methods.

Acknowledgments

This work was supported by the National Natural Science Foundation of China (Grant no. 62372315), Sichuan Science and Technology Planning Project (Grant no. 2024YFHZ0089 & 2024ZDZX0004), the

Chengdu Science and Technology Project (Grant no. 2023-XT00-00004-GX), and the Sichuan Science and Technology Miaozi Program (Grant no. MZGC20240057).

References

[1] Galen Andrew, Raman Arora, Jeff Bilmes, and Karen Livescu. 2013. Deep canonical correlation analysis. In *International conference on machine learning*. PMLR, 1247–1255.

[2] Mariusz Bojarski, Davide Del Testa, Daniel Dworakowski, Bernhard Firner, Beat Flepp, Prasoon Goyal, Lawrence D Jackel, Mathew Monfort, Urs Muller, Jiakai Zhang, et al. 2016. End to end learning for self-driving cars. *arXiv preprint arXiv:1604.07316* (2016).

[3] Bing Cao, Yinan Xia, Yi Ding, Changqing Zhang, and Qinghua Hu. 2024. Predictive Dynamic Fusion. In *International Conference on Machine Learning*. PMLR, 5608–5628.

[4] Cheng Deng, Zhaojia Chen, Xianglong Liu, Xinbo Gao, and Dacheng Tao. 2018. Triplet-based deep hashing network for cross-modal retrieval. *IEEE Transactions on Image Processing* 27, 8 (2018), 3893–3903.

[5] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. Bert: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 conference of the North American chapter of the association for computational linguistics: human language technologies, volume 1 (long and short papers)*. 4171–4186.

[6] Yu Geng, Zongbo Han, Changqing Zhang, and Qinghua Hu. 2021. Uncertainty-aware multi-view representation learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 35. 7545–7553.

[7] Wen Gu, Xiaoyan Gu, Jingzi Gu, Bo Li, Zhi Xiong, and Weiping Wang. 2019. Adversary guided asymmetric hashing for cross-modal retrieval. In *Proceedings of the 2019 on international conference on multimedia retrieval*. 159–167.

[8] Zongbo Han, Changqing Zhang, Huazhu Fu, and Joey Tianyi Zhou. 2020. Trusted multi-view classification. In *International Conference on Learning Representations*.

[9] Zongbo Han, Changqing Zhang, Huazhu Fu, and Joey Tianyi Zhou. 2022. Trusted multi-view classification with dynamic evidential fusion. *IEEE transactions on pattern analysis and machine intelligence* 45, 2 (2022), 2551–2566.

[10] Yadong Huo, Qin Qibing, Jiangyan Dai, Wenfeng Zhang, Lei Huang, and Chengduan Wang. 2024. Deep neighborhood-aware proxy hashing with uniform distribution constraint for cross-modal retrieval. *ACM Transactions on Multimedia Computing, Communications and Applications* 20, 6 (2024), 1–23.

[11] Yadong Huo, Qibing Qin, Jiangyan Dai, Lei Wang, Wenfeng Zhang, Lei Huang, and Chengduan Wang. 2023. Deep semantic-aware proxy hashing for multi-label cross-modal retrieval. *IEEE Transactions on Circuits and Systems for Video Technology* 34, 1 (2023), 576–589.

[12] Qing-Yuan Jiang and Wu-Jun Li. 2017. Deep cross-modal hashing. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 3232–3240.

[13] Chao Li, Cheng Deng, Ning Li, Wei Liu, Xinbo Gao, and Dacheng Tao. 2018. Self-supervised adversarial hashing networks for cross-modal retrieval. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 4242–4251.

[14] Ke Liang, Lingyuan Meng, Hao Li, Meng Liu, Siwei Wang, Sihang Zhou, Xinwang Liu, and Kunlun He. 2024. MGKsite: Multi-Modal Knowledge-Driven Site Selection via Intra and Inter-Modal Graph Fusion. *IEEE Transactions on Multimedia* (2024).

[15] Ke Liang, Lingyuan Meng, Hao Li, Jun Wang, Long Lan, Miaomiao Li, Xinwang Liu, and Huaimin Wang. 2025. From Concrete to Abstract: Multi-view Clustering on Relational Knowledge. *IEEE Transactions on Pattern Analysis and Machine Intelligence* (2025), 1–18. doi:10.1109/TPAMI.2025.3582689

[16] Ke Liang, Lingyuan Meng, Meng Liu, Yue Liu, Wenxuan Tu, Siwei Wang, Sihang Zhou, and Xinwang Liu. 2023. Learn from relational correlations and periodic events for temporal knowledge graph reasoning. In *Proceedings of the 46th international ACM SIGIR conference on research and development in information retrieval*. 1559–1568.

[17] Yijie Lin, Yuanbiao Gou, Xiaotian Liu, Jinfeng Bai, Jiancheng Lv, and Xi Peng. 2023. Dual contrastive prediction for incomplete multi-view representation learning. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 45, 4 (2023), 4447–4461.

[18] Wei Liu, Xiaodong Yue, Yufei Chen, and Thierry Denoeux. 2022. Trusted multi-view deep learning with opinion aggregation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 36. 7585–7593.

[19] Xiao Luo, Haixin Wang, Daqing Wu, Chong Chen, Minghua Deng, Jianqiang Huang, and Xian-Sheng Hua. 2023. A survey on deep hashing methods. *ACM Transactions on Knowledge Discovery from Data* 17, 1 (2023), 1–50.

[20] Richard J Perrin, Anne M Fagan, and David M Holtzman. 2009. Multimodal techniques for diagnosis and prognosis of Alzheimer's disease. *Nature* 461, 7266 (2009), 916–922.

[21] Xiaoshuang Shi, Manish Sapkota, Fuyong Xing, Fujun Liu, Lei Cui, and Lin Yang. 2018. Pairwise based deep ranking hashing for histopathology image classification and retrieval. *Pattern Recognition* 81 (2018), 14–22.

[22] Lingyun Song, Xuequn Shang, Chen Yang, and Mingxuan Sun. 2022. Attribute-guided multiple instance hashing network for cross-modal zero-shot hashing. *IEEE Transactions on Multimedia* 25 (2022), 5305–5318.

[23] Jing Sui, Shile Qi, Theo GM van Erp, Juan Bustillo, Rongtao Jiang, Dongdong Lin, Jessica A Turner, Eswar Damaraju, Andrew R Mayer, Yue Cui, et al. 2018.

Multimodal neuromarkers in schizophrenia via cognition-guided MRI fusion. *Nature communications* 9, 1 (2018), 3028.

[24] Yuan Sun, Xu Wang, Dezhong Peng, Zhenwen Ren, and Xiaobo Shen. 2023. Hierarchical hashing learning for image set classification. *IEEE Transactions on Image Processing* 32 (2023), 1732–1744.

[25] Wentao Tan, Lei Zhu, Jingjing Li, Huaxiang Zhang, and Junwei Han. 2022. Teacher-student learning: Efficient hierarchical message aggregation hashing for cross-modal retrieval. *IEEE Transactions on Multimedia* 25 (2022), 4520–4532.

[26] Junfeng Tu, Xueliang Liu, Zongxiang Lin, Richang Hong, and Meng Wang. 2022. Differentiable cross-modal hashing via multimodal transformers. In *Proceedings of the 30th ACM International Conference on Multimedia*. 453–461.

[27] Rong-Cheng Tu, Xian-Ling Mao, Rong-Xin Tu, Binbin Bian, Chengfei Cai, Hongfa Wang, Wei Wei, and Heyan Huang. 2022. Deep cross-modal proxy hashing. *IEEE Transactions on Knowledge and Data Engineering* 35, 7 (2022), 6798–6810.

[28] Huibing Wang, Mingze Yao, Guangqi Jiang, Zetian Mi, and Xianping Fu. 2023. Graph-collaborated auto-encoder hashing for multiview binary clustering. *IEEE Transactions on Neural Networks and Learning Systems* (2023).

[29] Huibing Wang, Mingze Yao, Guangqi Jiang, Zetian Mi, and Xianping Fu. 2024. Graph-Collaborated Auto-Encoder Hashing for Multiview Binary Clustering. *IEEE Transactions on Neural Networks and Learning Systems* 35, 7 (2024), 10121–10133.

[30] Weiran Wang, Raman Arora, Karen Livescu, and Jeff Bilmes. 2015. On deep multi-view representation learning. In *International conference on machine learning*. PMLR, 1083–1092.

[31] Jiwei Wei, Yang Yang, Xing Xu, Xiaofeng Zhu, and Heng Tao Shen. 2021. Universal weighting metric learning for cross-modal retrieval. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 44, 10 (2021), 6534–6545.

[32] Mengyao Xie, Zongbo Han, Changqing Zhang, Yichen Bai, and Qinghua Hu. 2023. Exploring and exploiting uncertainty for incomplete multi-view classification. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 19873–19882.

[33] Cai Xu, Jiajun Si, Ziyu Guan, Wei Zhao, Yue Wu, and Xiyue Gao. 2024. Reliable conflictive multi-view learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 38. 16129–16137.

[34] Cai Xu, Yilin Zhang, Ziyu Guan, and Wei Zhao. 2024. Trusted multi-view learning with label noise. *arXiv preprint arXiv:2404.11944* (2024).

[35] Xiaoqiang Yan, Shizhe Hu, Yiqiao Mao, Yangdong Ye, and Hui Yu. 2021. Deep multi-view learning methods: A review. *Neurocomputing* 448 (2021), 106–129.

[36] Mouxing Yang, Zhenyu Huang, Peng Hu, Taihao Li, Jiancheng Lv, and Xi Peng. 2022. Learning with twin noisy labels for visible-infrared person re-identification. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 14308–14317.

[37] Changqing Zhang, Yajie Cui, Zongbo Han, Joey Tianyi Zhou, Huazhu Fu, and Qinghua Hu. 2020. Deep partial multi-view learning. *IEEE transactions on pattern analysis and machine intelligence* 44, 5 (2020), 2402–2415.

[38] Changqing Zhang, Zongbo Han, Huazhu Fu, Joey Tianyi Zhou, Qinghua Hu, et al. 2019. CPM-Nets: Cross partial multi-view networks. *Advances in Neural Information Processing Systems* 32 (2019).

[39] Dongqing Zhang and Wu-Jun Li. 2014. Large-scale supervised multimodal hashing with semantic correlation maximization. In *Proceedings of the AAAI conference on artificial intelligence*, Vol. 28.

[40] Qingyang Zhang, Haitao Wu, Changqing Zhang, Qinghua Hu, Huazhu Fu, Joey Tianyi Zhou, and Xi Peng. 2023. Provable dynamic fusion for low-quality multimodal data. In *International conference on machine learning*. PMLR, 41753–41769.

[41] Zheng Zhang, Li Liu, Fumin Shen, Heng Tao Shen, and Ling Shao. 2018. Binary multi-view clustering. *IEEE transactions on pattern analysis and machine intelligence* 41, 7 (2018), 1774–1782.

[42] Jile Zhou, Guiguang Ding, and Yuchen Guo. 2014. Latent semantic sparse hashing for cross-modal similarity search. In *Proceedings of the 37th international ACM SIGIR conference on Research & development in information retrieval*. 415–424.