Contrastive Transformer Cross-Modal Hashing for Video-Text Retrieval

Xiaobo Shen[1] , **Qianxin Huang**[1] , **Long Lan**[2] and **Yuhui Zheng**[3*]
[1]Nanjing University of Science and Technology
[2]National University of Defense Technology
[3]Qinghai Normal University
njust.shenxiaobo@gmail.com, qianxinhuang@njust.edu.cn, long.lan@nudt.edu.cn,
zhengyh@vip.126.com

Abstract

As video-based social networks continue to grow exponentially, there is a rising interest in video retrieval using natural language. Cross-modal hashing, which learns compact hash code for encoding multi-modal data, has proven to be widely effective in large-scale cross-modal retrieval, e.g., image-text retrieval, primarily due to its computation and storage efficiency. However, when applied to video-text retrieval, existing cross-modal hashing methods generally extract features at the frame- or word-level for videos and texts individually, thereby ignoring their long-term dependencies. To address this issue, we propose Contrastive Transformer Cross-Modal Hashing (CTCH), a novel approach designed for video-text retrieval task. CTCH employs bidirectional transformer encoder to encode video and text and leverages their long-term dependencies. CTCH further introduces supervised multi-modality contrastive loss that effectively exploits inter-modality and intra-modality similarities among videos and texts. The experimental results on three video benchmark datasets demonstrate that CTCH outperforms the state-of-the-arts in video-text retrieval tasks.

1 Introduction

With the rapid development of the Internet, many video social networking sites and smartphone video sharing apps have emerged in recent years, and the number of videos on the Internet has witnessed an explosive growth. Efficient and accurate video retrieval is receiving ever-increasing attention. When searching with natural language, it is desirable to return relevant images and videos timely and accurately. As there is a big semantic gap among different modalities, e.g., video and text, it is challenging to perform video-text cross-modal search.

Hashing [Weiss *et al.*, 2008; Li *et al.*, 2020; Li *et al.*, 2022] has received a great deal of attention due to its efficient storage and computation over continuous features. The core idea of hashing is to project high-dimensional data points into compact hash codes, while preserving similarity among original data in the Hamming space. Cross-modal hashing maps multi-modal data into a common Hamming space, where efficient retrieval is performed. The shallow cross-modal hashing methods employ a two-stage strategy, that is, first extract the hand-crafted feature or deep feature using pre-trained network, and then learn hash code on the extracted feature of multi-modal data.

Deep cross-modal hashing [Wu *et al.*, 2019; Li *et al.*, 2022; Yu *et al.*, 2022] has been recently developed to perform latent hash code learning in an end-to-end manner. Conventional cross-modal hashing methods are primarily tailored for image-text retrieval, and they encounter two main challenges when extended to video-text retrieval. These methods typically extract features for each frame and word, pool them into video and sentence-level features, and are further trained to obtain hash codes. This overlooks correlations among the sequential units inherent in videos and sentences, and exploration of such correlation often helps improve performance. Some network structures, e.g., Recurrent Neural Network (RNN) [Lipton, 2015] and Long Short Term Memory (LSTM) [Hochreiter and Schmidhuber, 1997] can model sequence data. However, training LSTM is computationally expensive and also cannot well capture long-run dependencies among distant frames due to gradient vanishing [Pascanu *et al.*, 2013]. In addition, they are not sufficient to capture inter-modality and intra-modality similarity among video and text modalities due to their complex data structures. Therefore, it is still challenging to devise a cross-modal hashing specifically optimized for video-text retrieval.

To address the above concerns, we propose a novel cross-modal hashing method, i.e., Contrastive Transformer Cross-modal Hashing (CTCH) for video-text retrieval. CTCH applies a new video augmentation strategy and EDA [Wei and Zou, 2019] to augment video and text respectively. CTCH builds a network architecture that consists of bidirectional transformer encoder and a hash layer, and is mainly trained with multi-modality supervised contrastive loss. The overview of the proposed CTCH is illustrated in Figure 1. The main contributions of this work are summarized:

- We propose a novel cross-modal hashing method, i.e., Contrastive Transformer Cross-modal Hashing (CTCH) for video-text retrieval. CTCH is among the first attempts of cross-modal hashing that utilizes bidirectional

[*]Corresponding author.

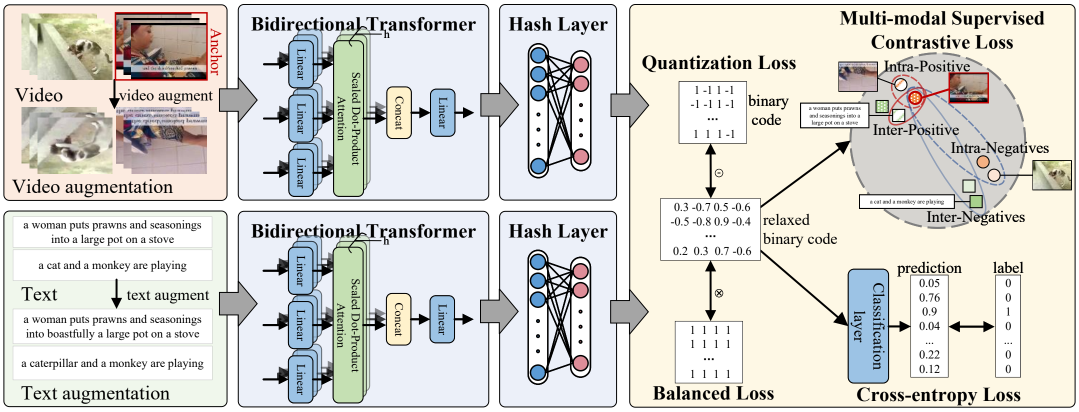


Figure 1: Illustration of the proposed CTCH. CTCH employs bidirectional transformer encoder to encode videos and texts, then constructs a hash layer to transform continuous features into hash code. CTCH devises multi-modality supervised contrastive loss, classification loss, quantization loss and balanced loss to supervise training.

transformer to effectively capture long-term dependencies in frame and word sequences.

- We devise a supervised multi-modality contrastive loss on constructed inter-modality and intra-modality triplet sets to effectively exploit structures among videos and texts at both inter-modality and intra-modality levels.
- The quantitative and qualitative results empirically verify the superiority of the proposed method over state-of-the arts in video-text retrieval.

2 Related Work

2.1 Cross-Modal Hashing

Cross-modal hashing aims to map multi-modal data, e.g., images, texts into a shared Hamming space, where hash codes of different modalities can be fast compared and matched. The primary advantage of cross-modal hashing is its efficiency in supporting efficient cross-modal retrieval, e.g., image-text retrieval. Existing cross-modal hashing methods includes two categories, i.e., shallow cross-modal hashing [Zhang and Li, 2014; Lin *et al.*, 2015] and deep cross-modal hashing [Jiang and Li, 2017; Li *et al.*, 2018; Qi *et al.*, 2021; Jin *et al.*, 2023], based on whether feature learning is performed by deep neural network.

The shallow cross-modal hashing methods learn linear function to transform multi-modal data into hash code. For instance, Semantic correlation maximization (SCM) [Zhang and Li, 2014] extends the canonical correlation analysis in the supervised manner. Semantics-preserving hashing (SePH) [Lin *et al.*, 2015] learns the joint binary hash codes by minimizing the Kullback–Leibler divergence between the hash codes and the semantic affinities, and then learns the cross-view hash functions with the kernel logistic regression. The shallow cross-modal hashing typically uses hand-crafted features, and feature learning is independent of hash code learning. Therefore, retrieval performance of shallow cross-modal hashing is expected to be further improved.

Deep cross-modal hashing leverages the power of deep learning, and integrates feature learning and hash code learning into a unified framework. Deep cross-modal hashing (DCMH) [Jiang and Li, 2017] performs an end-to-end learning framework, using a negative log-likelihood loss to preserve the cross-modal similarities. Self-Supervised Adversarial Hashing (SSAH) [Li *et al.*, 2018] utilizes two adversarial networks to jointly model different modalities and thereby further capture their semantic relevance and representation consistence under the supervision of the learned semantic feature. Existing cross-modal hashing methods primarily focus on image-text retrieval, while video-text retrieval is becoming increasingly important due to the richer information contained in videos. There are a few attempts [Qi *et al.*, 2021; Jin *et al.*, 2023] by encoding videos with I3D or LSTM. However, these methods often require high computational complexity during training. Developing effective cross-modal hashing for video-text retrieval still remains challenging due to complex structure of video.

2.2 Video Hashing

In recent years, some deep learning-based hashing methods have been proposed for video retrieval due to great achievement of deep neural networks in extracting high-level semantic features. Self-supervised Temporal Hashing (SSTH) [Zhang *et al.*, 2016] is a pioneer unsupervised video hashing method that models temporal sequence using LSTM. Specifically, SSTH applies Binary LSTM (BLSTM) to encode video to hash code, and minimizes reconstruction loss of deep autoencoder. Later, based on SSTH, Self-supervised Video Hashing (SSVH) [Song *et al.*, 2018] is further proposed by using hierarchical binary autoencoder and preserving similarity. Neighborhood Preserving Hashing (NPH) [Li *et al.*, 2019] preferentially encoded neighborhood-relevant visual content of a video into a binary code referring to pre-extracted neighborhood information. Bidirectional Transformer Hashing (BTH) [Li *et al.*, 2021] utilizes a bidirectional transformer as the backbone model and designs three self-supervised learning tasks to adequately capture the similarity structure in video data. However, these hashing methods are proposed for only video-video retrieval.

3 Contrastive Transformer Cross-modal Hashing

3.1 Problem Setup

Given a video-text dataset with N instances $\mathcal{O} = \{\mathcal{O}_i\}_{i=1}^N$, where $\mathcal{O}_i = \{\mathcal{V}_i, \mathcal{T}_i\}$, $\mathcal{V}_i$ and $\mathcal{T}_i$ are the corresponding video and text data of the i-th instance, CTCH aims to learn k-bit hash codes as compact representations for each video and text. We use CNN features $\{\mathbf{v}_i^m\}_{m=1}^M \in \mathbb{R}^{M \times d}$ of M sampled frames $\{\mathbf{f}_i^m\}_{m=1}^M$ to represent a video, where d is the dimension of each frame feature. For video modality, we feed to a bidirectional transformer based hash model to obtain binary codes $\mathbf{B}^v \in \{-1, 1\}^k$, where k is the code length. For text modality, we use a pre-trained tokenizer to convert text into token sequences, and then feed to a pre-trained bidirectional transformer based hash model to obtain binary codes $\mathbf{B}^t \in \{-1, 1\}^k$.

3.2 Data Augmentation

Video Augmentation

A popular video augmentation strategy is to apply random spatial augmentation approach, e.g., random cropping, color-jittering, blurring on each frame. However, such strategy can inevitably disrupt temporal structure, since consecutive frames may be augmented differently. In this work, we propose to apply the same spatial augmentation to frames. Moreover, we collect the augmented frames that do not overlap with original frames to capture a wider range of information. Algorithm 1 demonstrates the detailed procedure of temporally consistent spatial augmentation, where the parameters are first randomly generated for each video and then applied to all frames.

Text Augmentation

We employ the widely-used Easy Data Augmentation (EDA) [Wei and Zou, 2019] for text augmentation. Given a sentence, we apply four augmentation approaches randomly, including (1) Synonym Replacement (SR) that randomly replaces n non stop-words with synonyms, (2) Random Insertion (RI) that randomly inserts a synonym of a non stop-words into a random position, (3) Random Swap (RS) that randomly swaps two words, (4) Random Deletion (RD) that randomly removes each word with a probability.

3.3 Bidirectional Transformer Encoder

Inspired by the great success of self-attention in capturing correlations in a sequence [Vaswani *et al.*, 2017], we employ bidirectional transformer to encode video and text, both of which are essentially types of sequences.

Video Encoder

Bidirectional Transformer [Devlin *et al.*, 2019] has been shown powerful to process sequential data and successfully applied to various video applications [Sun *et al.*, 2019]. Some other sequential models, e.g., LSTM related networks, are limited by input sequence length and cannot well preserve long-term dependency in videos [Pascanu *et al.*, 2013]. In contrast, bidirectional transformer forms attention between any two frames in parallel, which is beneficial to model correlations between distant frames. This is the advantage over LSTM in video analysis. To exploit ordering information in the input sequence, we further use positional encoding in transformer. Following [Vaswani *et al.*, 2017], position feature $\mathbf{p}_{i,m}$ of the m-th frame from the i-th video is generated by sine and cosine functions of different frequencies: $(\mathbf{p}_{i,m})_{2j} = \sin\left(m/10000^{2j/d}\right)$, $(\mathbf{p}_{i,m})_{2j+1} = \cos\left(m/10000^{2j/d}\right)$, where j denotes index of dimension. The position embedding and visual sequence are of equal length, and are first added and then fed into video transformer. Assume there are L transformer layers, and each layer is constructed by multi-head attention [Vaswani *et al.*, 2017]. Specifically, in each transformer layer, given an input sequence of embedding $\mathbf{X}$, the j-th attention head projects $\mathbf{X}$ to a triplet of (query, key, value) denoted as ($\mathbf{Q}_j$, $\mathbf{K}_j$, $\mathbf{V}_j$) via three learnable parameters, i.e., $\mathbf{W}_j^q$, $\mathbf{W}_j^k$, and $\mathbf{W}_j^v$ respectively. A scaled dot-product attention is applied between $\mathbf{Q}_j$ and $\mathbf{K}_j$, and its output is then fed to softmax function to obtain attentional distribution over $\mathbf{V}_j$. Formally, the j-th attention head computes the output embedding sequence as follows:

$$\mathbf{A}_j = \text{softmax}\left(\frac{\mathbf{Q}_j \mathbf{K}_j^\top}{\sqrt{d_k}}\right)\mathbf{V}_j \tag{1}$$

where $\mathbf{Q}_j = \mathbf{X}\mathbf{W}_j^q$, $\mathbf{K}_j = \mathbf{X}\mathbf{W}_j^k$, $\mathbf{V}_j = \mathbf{X}\mathbf{W}_j^v$, d_k is a scaling factor. After being passed through L transformer layers, these input tokens are mapped to a sequence of l-D latent visual embeddings $\{\mathbf{h}_{i,m}^v\}_{m=1}^M$. Each of these embeddings contains not only the visual content in corresponding frame, but also information flowing from other frames in both direction within the video.

Algorithm 1 Video augmentation

```
**Crop**($f$): Crop $f$ with a random size;
**Resize**($f$): Resize $f$ to size of $224 \times 224$;
**Flip**($f$): Flip $f$ horizontally or vertically randomly;
**Noise**($f$): Add Gaussian noise to $f$;
**Gray**($f$): Convert $f$ to grayscale with a probability of 0.2;
**Blur**($f$): Apply Gaussian blur to $f$ with a probability of 0.8;
**Input**: Video clip $\mathcal{V} = \{\mathbf{f}^m\}_{m=1}^M$;
**Output**: $\hat{\mathcal{V}} = \{\hat{\mathbf{f}}^m\}_{m=1}^M$.
Randomly generate augmentation parameters;
for $m \in \{1 \dots M\}$ do
    $\hat{\mathbf{f}}^m$ = Resize(Crop($\mathbf{f}^m$))
    $\hat{\mathbf{f}}^m$ = Flip($\hat{\mathbf{f}}^m$)
    $\hat{\mathbf{f}}^m$ = Noise($\hat{\mathbf{f}}^m$)
    $\hat{\mathbf{f}}^m$ = Gray($\hat{\mathbf{f}}^m$)
    $\hat{\mathbf{f}}^m$ = Blur($\hat{\mathbf{f}}^m$)
end for
```

Text Encoder

The bidirectional transformer facilitates deeper understanding of context by allowing simultaneous processing of input data from both directions. This feature makes it highly effective at capturing the interdependencies between words in a sentence, thereby enhancing the performance of NLP tasks. With such superiority, this work employs bidirectional transformer to encode text. The tokenizer first splits a text corresponding to a video into words, and truncates them to a unified length. Each word is assigned to a unique token in vocabulary table, and each sentence can be transformed into a token sequence. Particularly, special classification token ([CLS]) and unknown token ([UNK]) are inserted to indicate the beginning of a sentence and out-of-vocabulary word respectively. In each sentence, we convert each token into an embedding, aggregate embeddings of all the tokens, and feed the aggregation to the bidirectional transformer. Text encoder has the same structure with video encoder.

Hash Layer

The representation obtained through the transformer is real-valued, a hash layer is utilized to project the continuous representation into discrete hash code. Retrieval can be more efficient and storage can be saved to a large extent by this way. Taking video as an example, we project $\{\mathbf{h}^v_{i,m}\}_{m=1}^M$ to a sequence of real-valued vectors $\{\mathbf{z}^v_{i,m}\}_{m=1}^M$ via a Fully Connected (FC) layer. It can be formulated as:

$$\{\mathbf{z}^v_{i,m}\}_{m=1}^M = \mathrm{FC}\left(\{\mathbf{h}^v_{i,m}\}_{m=1}^M, k\right) \tag{2}$$

where FC$(\cdot, k)$ is mapping function of the FC layer that maps original feature into k-bit hash code. The mean pooling and quantization are applied on frame-level feature to obtain video-level hash code, which can be denoted as:

$$\mathbf{z}^v_i = \tanh\left(\frac{1}{M}\sum_{m=1}^{M}\mathbf{z}^v_{i,m}\right), \quad \mathbf{b}^v_i = \mathrm{sgn}\left(\mathbf{z}^v_i\right) \tag{3}$$

where sgn is sign function, and $\mathrm{sgn}(x) = 1$ if $x \geq 0$ and $\mathrm{sgn}(x) = -1$ otherwise. Accordingly, the continuous feature $\mathbf{z}^t_i$ and hash code $\mathbf{b}^t_i$ in text modality can be obtained as follows:

$$\mathbf{z}^t_i = \mathrm{FC}\left(\mathbf{h}^t_i, k\right), \quad \mathbf{b}_i = \mathrm{sgn}\left(\mathbf{z}_i\right) \tag{4}$$

where $\mathbf{h}^t_i$ denotes the i-th latent text embedding.

3.4 Multi-modal Supervised Contrastive Learning

As a dominant component in self-supervised learning, contrastive learning has received increasing interests due to its success in many research areas, e.g.,computer vision [Qian *et al.*, 2021], multimedia analysis [Chen *et al.*, 2021]. Contrastive learning [Chen *et al.*, 2020] is based on a triplet set consisting of anchor, positive, and negative samples. The goal of contrastive learning is to pull anchor and positive sample together and push apart anchor from negative samples. This work considers the benefits of contrastive learning in learning hash codes as compact representation for video and text.

Given a set of N samples $\{\mathbf{x}_i, \mathbf{g}_i\}_{i=1}^N$, $\mathbf{g}_i$ is the label vector of $\mathbf{x}_i$. Its augmentated set is denoted as $\{\hat{\mathbf{x}}_i, \mathbf{g}_i\}_{i=1}^{2N}$ that has $2N$ samples, and $\hat{\mathbf{x}}_{2i-1}$ and $\hat{\mathbf{x}}_{2i}$ are two random augmentations of $\mathbf{x}_i (i = 1 \ldots N)$. Assume that $i \in I = \{1 \ldots 2N\}$ be the index of an arbitrary augmented sample, and $(\hat{\mathbf{x}}_i, \hat{\mathbf{x}}_j)$ be a positive pair. The conventional self-supervised contrastive loss [Chen *et al.*, 2020] is defined as follow:

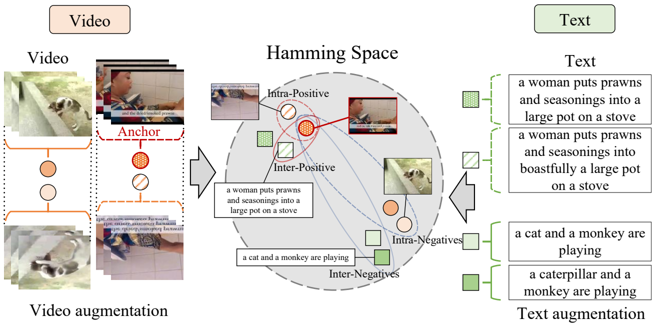

Video
Text
Video
Anchor
Video augmentation
Hamming Space
Intra-Positive
Inter-Positive
a woman puts prawns and seasonings into a large pot on a stove
a cat and a monkey are playing
Intra-Negatives
Inter-Negatives
Text
a woman puts prawns and seasonings into a large pot on a stove
a woman puts prawns and seasonings into boastfully a large pot on a stove
a cat and a monkey are playing
a caterpillar and a monkey are playing
Text augmentation

Figure 2: Illustration of multi-modality contrastive learning. We construct inter- and intra-modality triplet sets. Based on the two sets, we define inter- and intra-modality contrastive losses to preserve inter- and intra-modality similarity structures respectively.

$$\mathbf{L}_{self} = -\sum_{i=1}^{2N}\log\frac{\exp(\hat{\mathbf{x}}_i \cdot \hat{\mathbf{x}}_j/\tau)}{\sum\limits_{a\in\mathbf{A}(i)}\exp(\hat{\mathbf{x}}_i \cdot \hat{\mathbf{x}}_a/\tau)} \tag{5}$$

where τ is a temperature coefficient that controls dynamic range of product, $\mathbf{A}(i) = \{a|a \in I, a \neq i\}$, $\hat{\mathbf{x}}_i$ and $\hat{\mathbf{x}}_j$ are called the anchor and positive respectively, and the other $2(N-1)$ samples $\{\hat{\mathbf{x}}_k|\hat{\mathbf{x}}_k \in I, k \neq i, k \neq j\}$ are called the negatives. As label information is not incorporated in self-supervised constrastive loss, its performance is limited.

We propose supervised multi-modal contrastive loss that fully considers label supervision information to improve representation capability of video and text modalities. As shown in Figure 2, the proposed multi-modal supervised contrastive loss includes inter- and intra-modality contrastive losses to preserve inter- and intra-modality similarity structure respectively.

Inter-modality Contrastive Loss

Inter-modality contrastive learning is defined based on a triplet set of anchor, inter-positive, and inter-negative samples. Inter-modality contrastive learning simultaneously encourages embedding of anchor to be close to that of inter-positive sample and to be far away from those of inter-negative samples, such that cross-modality correlation can be effectively exploited in latent embedding space.

Let $\mathbf{z}^v_i$ and $\hat{\mathbf{z}}^v_i$ be original and augmented features of the i-th video. Similarly, $\mathbf{z}^t_i$ and $\hat{\mathbf{z}}^t_i$ is the original and augmented feature of i-th text. In this work, we define a positive index set as $\mathbf{P}(i) = \{p|p \in \mathbf{A}(i), \mathbf{g}_p = \mathbf{g}_i\}$, $|\mathbf{P}(i)|$ denotes cardinality of $\mathbf{P}(i)$. By regarding $\mathbf{z}^v_i$ and $\hat{\mathbf{z}}^v_i$ as anchors, we have the following inter-modality contrastive loss:

$$\begin{aligned}\mathbf{L}^{ve}_m = &\sum_{i=1}^{N}\frac{1}{|\mathbf{P}(i)|}\sum_{p\in\mathbf{P}(i)} -\log\frac{\exp(\mathbf{z}^v_i \cdot \mathbf{z}^t_p/\tau)}{\sum\limits_{a\in\mathbf{A}(i)}\exp(\mathbf{z}^v_i \cdot \mathbf{z}^t_a/\tau)} \\ &+\sum_{i=1}^{N}\frac{1}{|\mathbf{P}(i)|}\sum_{p\in\mathbf{P}(i)} -\log\frac{\exp(\hat{\mathbf{z}}^v_i \cdot \hat{\mathbf{z}}^t_p/\tau)}{\sum\limits_{a\in\mathbf{A}(i)}\exp(\hat{\mathbf{z}}^v_i \cdot \hat{\mathbf{z}}^t_a/\tau)}\end{aligned} \tag{6}$$

(6) is inspired by SupCon [Khosla *et al.*, 2020], and considers

Algorithm 2 Contrastive Transformer Cross-modal Hashing

```
Input: video-text pairs $\mathcal{O}_i = \{\mathcal{V}_i, \mathcal{T}_i\}_{i=1}^N$; input dimension $d$; code length $k$; batch size; number of epochs; learning rate; constants $\alpha$, $\beta$, $\gamma$, $d_k$, $\tau$.
Output: Network parameters.
for each epoch do
    for each iteration do
        Sample a minibatch randomly;
        Obtain $\{\mathbf{h}_{i,m}^v\}_{m=1}^M$ and $\{\mathbf{h}_i^t\}$ via bidirectional transformer;
        Obtain $\mathbf{z}_i^v$, $\mathbf{b}_i^v$ , $\mathbf{z}_i^t$, $\mathbf{b}_i^t$ via (2), (3) and (4);
        Calculate multi-modality supervised contrastive loss $\mathcal{L}_m$ via (8);
        Update network parameters to minimize (13) via BP algorithm;
    end for
end for
```

all the samples belonging to the same class as positives. Minimizing the first term in (6) enables embeddings among original videos and texts in the same class to have high similarities. Similarly, the second term applies to augmented videos and texts. Accordingly, inter-modality contrastive loss $\mathcal{L}_m^{te}$ for text can be similarly defined.

Intra-modality Contrastive Loss

Intra-modality contrastive learning defines a triplet set of anchor, intra-positive, and intra-negative samples. Intra-modality contrastive learning enables embedding of anchor to be close to that of intra-positive sample and far away from those of intra-negative samples, and thus preserves intrinsic similarity structure within each modality. Taking video modality as example, the intra-modality contrastive loss is defined as follows:

$$\mathbf{L}_m^{va} = \sum_{i=1}^{N} \frac{1}{|\mathbf{P}(i)|} \sum_{p\in\mathbf{P}(i)} -\log \frac{\exp(\mathbf{z}_i^v \cdot \hat{\mathbf{z}}_p^v/\tau)}{\sum\limits_{a\in\mathbf{A}(i)} \exp(\mathbf{z}_i^v \cdot \hat{\mathbf{z}}_a^v/\tau)} \tag{7}$$

Accordingly, intra-modality contrastive loss $\mathcal{L}_m^{ta}$ for text modality can be similarly defined.

To sum up, we have the following multi-modal supervised contrastive learning loss:

$$\mathcal{L}_m = \mathcal{L}_m^{va} + \mathcal{L}_m^{ta} + \mathcal{L}_m^{ve} + \mathcal{L}_m^{te} \tag{8}$$

3.5 Overall Objective

Classification Loss

To fully exploit label information, the learned hash codes are directly used for classification. Taking video modality as example, we consider the widely-used cross-entropy loss:

$$\mathcal{L}_c^v = -\frac{1}{N} \sum_{i=1}^{N} \sum_{c=1}^{C} \mathbf{g}_{ic}^v \log \hat{\mathbf{g}}_{ic}^v - (1 - \mathbf{g}_{ic}^v) \log(1 - \hat{\mathbf{g}}_{ic}^v) \tag{9}$$

where $\hat{\mathbf{g}}_c^v = \phi(\mathrm{FC}(\mathbf{b}_i^v))$ denotes label prediction using hash code of the i-th video, and ϕ is sigmoid function, $\mathbf{g}_{ic}^v$ is the c-th element in $\mathbf{g}_i^v$. The final classification loss in both text and video modalities is defined as follows:

$$\mathcal{L}_c = \mathcal{L}_c^v + \mathcal{L}_c^t \tag{10}$$

Quantization Loss

As the network output is not binary, direct binarization will lead to large quantization error. Thus it is encouraging to enable the output close to $[-1, +1]$ to control quantization error. Based on such idea, we propose the following quantization loss which is based on the bi-modal Laplacian prior:

$$\mathcal{L}_q = \||\mathbf{Z}^v| - 1\|_1 + \||\mathbf{Z}^t| - 1\|_1 \tag{11}$$

As the derivative of absolute function is difficult to compute, we instead apply its smooth surrogate, i.e., $|x| \approx \log\cosh x$.

Balanced Loss

To enable hash code to be balanced, hashing enforces each bit to be mean-zero [Yang *et al.*, 2018]. We define the balance loss as follow:

$$\mathcal{L}_b = ||\mathbf{Z}^v\mathbf{1}||_F^2 + ||\mathbf{Z}^t\mathbf{1}||_F^2 \tag{12}$$

Total Loss

To this end, the overall objective function of the proposed CTCH is defined as follows:

$$\mathcal{L} = \mathcal{L}_m + \alpha\mathcal{L}_c + \beta\mathcal{L}_q + \gamma\mathcal{L}_b \tag{13}$$

where α, β, γ are trade-off parameters that control relative importances of four losses. The backpropagation algorithm is used to optimize network. The training procedure of the proposed CTCH is illustrated in Algorithm 2.

4 Experiment

4.1 Datasets

The experiments are conducted on three benchmark video text datasets, which have been widely used for video-text analysis. The three datasets are detailed as follows:

MSR-VTT [Xu *et al.*, 2016] is the largest general video captioning dataset. It contains 10,000 video clips with 41.2 hours and 200,000 clip-sentence pairs in 20 categories, and 20 natural sentences annotated manually for each video clip. Following [Xu *et al.*, 2016], we randomly choose 6,513 and 2,990 clips for training and testing respectively.

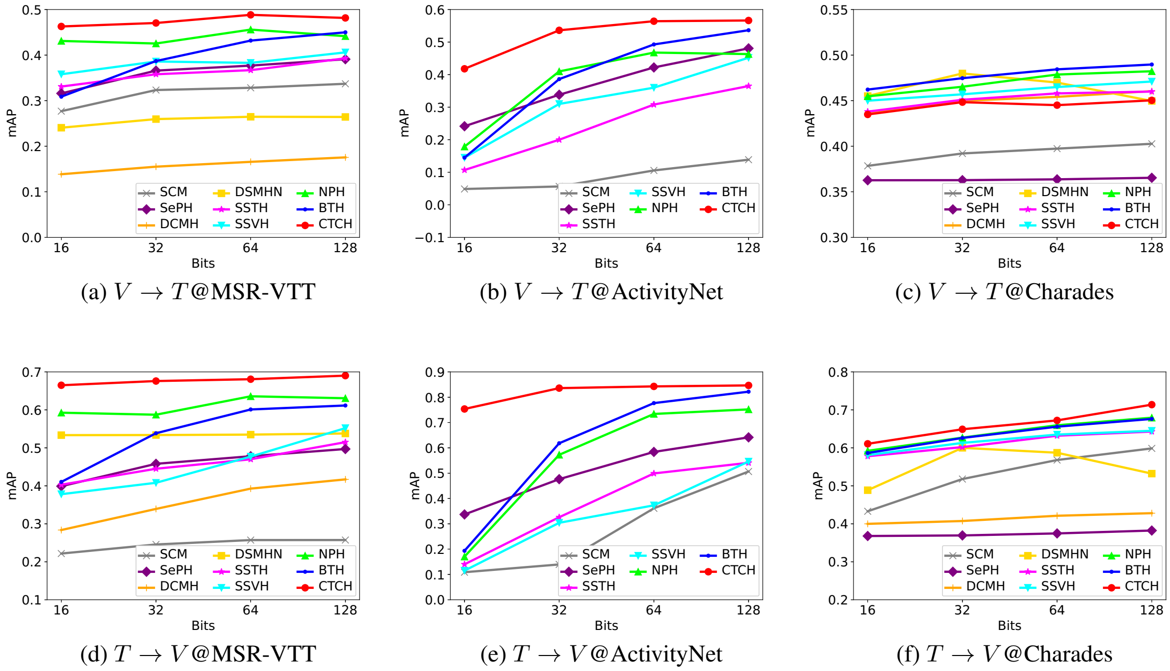
ActivityNet Captions v1.2 [Krishna *et al.*, 2017] is a large-scale video dataset for human action understanding. It contains more than 13,000 videos from 100 activity categories collected from YouTube, with an average of 137 untrimmed videos per class and 1.41 activity instances per video. We randomly choose 4,816 and 2,382 videos for training and testing respectively.

Charades [Sigurdsson *et al.*, 2016] is a dataset composed of 9848 videos of daily indoors activities collected through Amazon Mechanical Turk. The dataset contains 66,500 temporal annotations for 157 action classes, 41,104 labels for 46 object classes, and 27,847 textual descriptions of the videos. Since test set does not provide labels, we use validation set for testing. We choose 7985 and 1863 videos for training and testing respectively.

4.2 Experiment Setting

Baselines

To our knowledge, there are few hashing methods specifically designed for video-text retrieval. We compare CTCH with



(a) $V \rightarrow T$@MSR-VTT

(b) $V \rightarrow T$@ActivityNet

(c) $V \rightarrow T$@Charades

(d) $T \rightarrow V$@MSR-VTT

(e) $T \rightarrow V$@ActivityNet

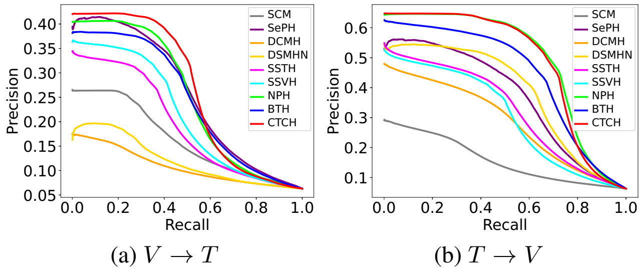
(f) $T \rightarrow V$@Charades

Figure 3: mAPs of all the methods on three video-text retrieval tasks on three benchmark datasets.

eight state-of-the-art methods, including two shallow cross-modal hashing methods, i.e., SCM [Zhang and Li, 2014], SePH [Lin *et al.*, 2015], two deep cross-modal hashing methods, i.e., DCMH [Jiang and Li, 2017], DSMHN [Jin *et al.*, 2023], and four deep video hashing methods, i.e., SSTH [Zhang *et al.*, 2016], SSVH [Song *et al.*, 2018], NPH [Li *et al.*, 2019] and BTH [Li *et al.*, 2021]. For SCM and SePH, following [Qi *et al.*, 2021], we mean pool features extracted by VGG-16 to represent video. For DCMH and DSMHN, we use I3D [Carreira and Zisserman, 2017] as backbone to encode video. SSTH, SSVH, NPH, and BTH are developed only for video, we use the same text encoder as DCMH, and train them using loss of DCMH.

Setup

Following [Zhang *et al.*, 2016], we first sample 25 frames resized to 224×224 for each video, and extract 4096-D frame features with VGG-16 [Simonyan and Zisserman, 2015] pre-trained on ImageNet [Russakovsky *et al.*, 2015]. Video transformer includes four layers with 256-D attention head, and the scaling factor d_k is set to 256. We first concatenate all texts belonging to the same video, then tokenize concatenated text as input. Text transformer has the same structure as video transformer, and its pre-trained model is provided by Hugging Face. The batch size, number of epochs, and learning rate are set to 256, 200, and 1×10^{-4} respectively. The parameters α, β, and γ are set to 400, 0.05 and 0.05 respectively. The temperature coefficient is set to 0.2. The proposed CTCH is optimized using Adam optimizer.



(a) $V \rightarrow T$

(b) $T \rightarrow V$

Figure 4: PR curves with 64-bit hash code on MST-VTT.

Evaluation Metrics

Following [Zhang *et al.*, 2016], we consider the widely-used mean Average Precision (mAP) and Precision-Recall (PR) curve as evaluation metrics [Over *et al.*, 2010].

4.3 Comparisons with State-of-the-arts

Figure 3 shows the mAPs of all the methods on two video-text retrieval tasks on three benchmark datasets. In addition, the PR curves with 64 bits on MSR-VTT are shown in Figure 4. DCMH and DSMHN use I3d as backbone, and require very large batch size for training on ActivityNet, which has 100 categories. Therefore, the results of the two methods on ActivityNet are not reported. From these results, we have the following interesting observations:

- The proposed CTCH has best mAPs in most cases. For instance, on MSR-VTT, CTCH improves the best baseline, i.e., NPH by 3.18%, 4.16%, 2.78% and 4% with 16,

Method	$V \to T$		$T \to V$	
	32 bits	64 bits	32 bits	64 bits
CTCH-Cla	0.4519	0.4664	0.6422	0.6576
CTCH-InterC	0.1186	0.1616	0.1337	0.1283
CTCH-IntraC	0.4592	0.4689	0.6492	0.6563
CTCH-Aug1	0.4695	0.4739	0.6442	0.6546
CTCH-Aug2	0.4552	0.4729	0.6488	0.6578
CTCH-Aug3	**0.4716**	0.4845	0.6591	0.6637
CTCH	0.4705	**0.4884**	**0.676**	**0.6808**

Table 1: mAPs of ablation study in terms of loss and video augmentation on MSR-VTT.

32, 64 and 128 bits respectively for $V \to T$ task, and by 7.23%, 9.78%, 5.16% and 5.95% with 16, 32, 64 and 128 bits respectively for $T \to V$ task. The PR curves of CTCH are higher than those of the baselines.

- Deep hashing methods outperform shallow hashing methods in most cases. Among deep methods, BTH and NPH exhibit the best performances, showcasing superior capabilities of bidirectional transformer and LSTM in capturing long-term dependencies in sequences.
- Video hashing methods outperform image hashing methods. This indicates that for video-text retrieval tasks, video hashing can extract more useful semantic information from the videos than image cross-modal hashing.

4.4 Ablation Study

Analysis on Loss

We compare four variants of the proposed method with different losses, including (1) CTCH-Cla: a variant that removes classification loss; (2) CTCH-InterC: a variant that removes inter-modal contrastive loss; (3) CTCH-IntraC: a variant that removes intra-modal contrastive loss. Table 1 reports mAPs of the variants on two retrieval tasks on MSR-VTT. As can be observed, CTCH significantly outperforms CTCH-InterC, demonstrating the importance of inter-modality contrastive loss. Meanwhile, CTCH further improves CTCH-IntraC and CTCH-Cla, verifying the effectiveness of intra-modality contrastive and classification losses.

Analysis on Augmentation Strategy

We compare three variants of the proposed method using different augmentation strategies, including (1) CTCH-Aug1: a variant that chooses non-overlapping frames and apply random augmentation; (2) CTCH-Aug2: a variant that chooses the same frames and apply uniform augmentation; (3) CTCH-Aug3: a variant that chooses non-overlapping frames without augmentation. From Table 1, we see that the proposed CTCH outperforms the three variants. It shows the augmentation strategy in the proposed method that uniformly augments non-overlapping frames allows to capture more information, providing better retrieval performance.

4.5 Parameter Analysis

We study the sensitivity of one trade-off parameter, i.e., α in the proposed method, which are varied from $[1, 5000]$. Figure 5 shows the mAPs results with respect to different α with

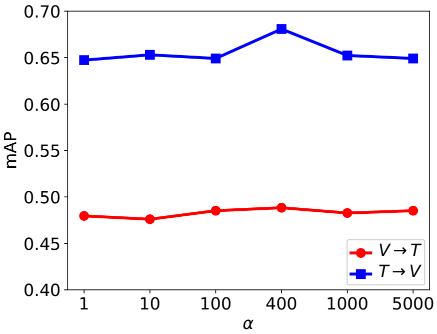
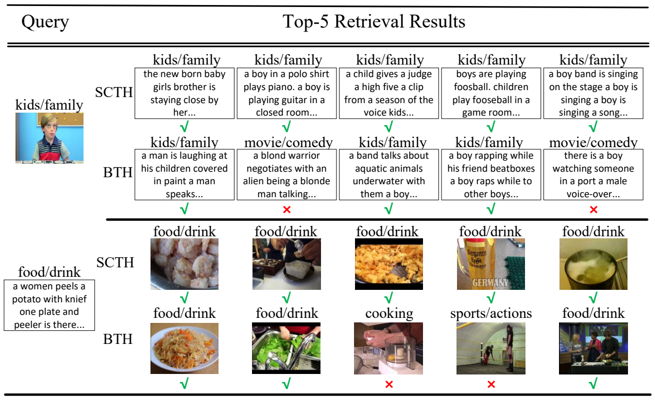


Figure 5: mAPs of CTCH with different α on MST-VTT.



the new born baby girls brother is staying close by her...

a boy in a polo shirt plays piano. a boy is playing guitar in a closed room...

a child gives a judge a high five a clip from a season of the voice kids...

boys are playing foosball. children play fooseball in a game room...

a boy band is singing on the stage a boy is singing a boy is singing a song...

a man is laughing at his children covered in paint a man speaks...

a blond warrior negotiates with an alien being a blonde man talking...

a band talks about aquatic animals underwater with them a boy...

a boy rapping while his friend beatboxes a boy raps while to other boys...

there is a boy watching someone in a port a male voice-over...

a women peels a potato with knief one plate and peeler is there...

Figure 6: Top-5 retrieved results of BTH and the proposed CTCH on one randomly selected query video and text from MSR-VTT. The correct and incorrect retrieved results are marked by tick and cross respectively.

64-bit hash code on MSR-VTT. We clearly see that CTCH remains relatively stable with change of α.

4.6 Case Study

This section presents case study on video-text retrieval. Figure 6 illustrates the top-5 results retrieved of the proposed CTCH and the competitive BTH on MSR-VTT. From this figure, we see that the proposed CTCH is capable of retrieving more correct results than BTH.

5 Conclusion

In this work, we propose a new cross-modal hashing method designed for video-text retrieval. CTCH effectively explores long-term dependencies among video frames and text words via bidirectional transformer. CTCH exploits inter-modality and intra-modality similarities among videos and texts by minimizing multi-modality supervised contrastive loss. Extensive empirical results demonstrate the superiority of the proposed method and verify the effectiveness of each component. In the future, we consider learning hash code on incomplete video and text data.

Acknowledgments

This work was supported by the National Natural Science Foundation of China under Grant No. 62176126, U20B2065, U22B2056, 62272468, the Natural Science Foundation of Jiangsu Province, China under Grant No. BK20230095, BK20211539.

References

[Carreira and Zisserman, 2017] João Carreira and Andrew Zisserman. Quo vadis, action recognition? A new model and the kinetics dataset. In *CVPR*, pages 4724–4733, 2017.

[Chen *et al.*, 2020] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey E. Hinton. A simple framework for contrastive learning of visual representations. In *ICML*, volume 119, pages 1597–1607, 2020.

[Chen *et al.*, 2021] Yanbei Chen, Yongqin Xian, A. Sophia Koepke, Ying Shan, and Zeynep Akata. Distilling audio-visual knowledge by compositional contrastive learning. In *CVPR*, pages 7016–7025, 2021.

[Devlin *et al.*, 2019] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: pre-training of deep bidirectional transformers for language understanding. In *NAACL*, pages 4171–4186, 2019.

[Hochreiter and Schmidhuber, 1997] Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural Computation*, 9(8):1735–1780, 1997.

[Jiang and Li, 2017] Qing-Yuan Jiang and Wu-Jun Li. Deep cross-modal hashing. In *CVPR*, pages 3270–3278, 2017.

[Jin *et al.*, 2023] Lu Jin, Zechao Li, and Jinhui Tang. Deep semantic multimodal hashing network for scalable image-text and video-text retrievals. *IEEE TNNLS*, 34(4):1838–1851, 2023.

[Khosla *et al.*, 2020] Prannay Khosla, Piotr Teterwak, Chen Wang, Aaron Sarna, Yonglong Tian, Phillip Isola, Aaron Maschinot, Ce Liu, and Dilip Krishnan. Supervised contrastive learning. *NeurIPS*, 33:18661–18673, 2020.

[Krishna *et al.*, 2017] Ranjay Krishna, Kenji Hata, Frederic Ren, Li Fei-Fei, and Juan Carlos Niebles. Dense-captioning events in videos. In *ICCV*, pages 706–715, 2017.

[Li *et al.*, 2018] Chao Li, Cheng Deng, Ning Li, Wei Liu, Xinbo Gao, and Dacheng Tao. Self-supervised adversarial hashing networks for cross-modal retrieval. In *CVPR*, pages 4242–4251, 2018.

[Li *et al.*, 2019] Shuyan Li, Zhixiang Chen, Jiwen Lu, Xiu Li, and Jie Zhou. Neighborhood preserving hashing for scalable video retrieval. In *ICCV*, pages 8211–8220, 2019.

[Li *et al.*, 2020] Zechao Li, Jinhui Tang, Liyan Zhang, and Jian Yang. Weakly-supervised semantic guided hashing for social image retrieval. *IJCV*, 128(8):2265–2278, 2020.

[Li *et al.*, 2021] Shuyan Li, Xiu Li, Jiwen Lu, and Jie Zhou. Self-supervised video hashing via bidirectional transformers. In *CVPR*, pages 13549–13558, 2021.

[Li *et al.*, 2022] Shuyan Li, Xiu Li, Jiwen Lu, and Jie Zhou. Structure-adaptive neighborhood preserving hashing for scalable video search. *IEEE TCSVT*, 32(4):2441–2454, 2022.

[Lin *et al.*, 2015] Zijia Lin, Guiguang Ding, Mingqing Hu, and Jianmin Wang. Semantics-preserving hashing for cross-view retrieval. In *CVPR*, pages 3864–3872, 2015.

[Lipton, 2015] Zachary Chase Lipton. A critical review of recurrent neural networks for sequence learning. *CoRR*, abs/1506.00019, 2015.

[Over *et al.*, 2010] Paul Over, George Awad, Jon Fiscus, Brian Antonishek, Martial Michel, Wessel Kraaij, Alan F. Smeaton, and Georges Quénot. TRECVID 2010 - an overview of the goals, tasks, data, evaluation mechanisms and metrics. In *TRECVID 2010 workshop participants notebook papers*, 2010.

[Pascanu *et al.*, 2013] Razvan Pascanu, Tomás Mikolov, and Yoshua Bengio. On the difficulty of training recurrent neural networks. In *ICML*, volume 28, pages 1310–1318, 2013.

[Qi *et al.*, 2021] Mengshi Qi, Jie Qin, Yi Yang, Yunhong Wang, and Jiebo Luo. Semantics-aware spatial-temporal binaries for cross-modal video retrieval. *IEEE TIP*, 30:2989–3004, 2021.

[Qian *et al.*, 2021] Rui Qian, Tianjian Meng, Boqing Gong, Ming-Hsuan Yang, Huisheng Wang, Serge J. Belongie, and Yin Cui. Spatiotemporal contrastive video representation learning. In *CVPR*, pages 6964–6974, 2021.

[Russakovsky *et al.*, 2015] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael S. Bernstein, Alexander C. Berg, and Li Fei-Fei. Imagenet large scale visual recognition challenge. *IJCV*, 115(3):211–252, 2015.

[Sigurdsson *et al.*, 2016] Gunnar A. Sigurdsson, Gül Varol, Xiaolong Wang, Ali Farhadi, Ivan Laptev, and Abhinav Gupta. Hollywood in homes: Crowdsourcing data collection for activity understanding. In *ECCV*, volume 9905, pages 510–526, 2016.

[Simonyan and Zisserman, 2015] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. In *ICLR*, 2015.

[Song *et al.*, 2018] Jingkuan Song, Hanwang Zhang, Xiangpeng Li, Lianli Gao, Meng Wang, and Richang Hong. Self-supervised video hashing with hierarchical binary auto-encoder. *IEEE TIP*, 27(7):3210–3221, 2018.

[Sun *et al.*, 2019] Chen Sun, Austin Myers, Carl Vondrick, Kevin Murphy, and Cordelia Schmid. Videobert: A joint model for video and language representation learning. In *ICCV*, pages 7463–7472, 2019.

[Vaswani *et al.*, 2017] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *NeurIPS*, pages 5998–6008, 2017.

[Wei and Zou, 2019] Jason W. Wei and Kai Zou. EDA: easy data augmentation techniques for boosting performance on text classification tasks. In *EMNLP*, pages 6381–6387, 2019.

[Weiss *et al.*, 2008] Yair Weiss, Antonio Torralba, and Robert Fergus. Spectral hashing. In *NeurIPS*, pages 1753–1760, 2008.

[Wu *et al.*, 2019] Gengshen Wu, Jungong Han, Yuchen Guo, Li Liu, Guiguang Ding, Qiang Ni, and Ling Shao. Unsupervised deep video hashing via balanced code for large-scale video retrieval. *IEEE TIP*, 28(4):1993–2007, 2019.

[Xu *et al.*, 2016] Jun Xu, Tao Mei, Ting Yao, and Yong Rui. MSR-VTT: A large video description dataset for bridging video and language. In *CVPR*, pages 5288–5296, 2016.

[Yang *et al.*, 2018] Huei-Fang Yang, Kevin Lin, and Chu-Song Chen. Supervised learning of semantics-preserving hash via deep convolutional neural networks. *IEEE TPAMI*, 40(2):437–451, 2018.

[Yu *et al.*, 2022] En Yu, Jianhua Ma, Jiande Sun, Xiaojun Chang, Huaxiang Zhang, and Alexander G. Hauptmann. Deep discrete cross-modal hashing with multiple supervision. *Neurocomputing*, 486:215–224, 2022.

[Zhang and Li, 2014] Dongqing Zhang and Wu-Jun Li. Large-scale supervised multimodal hashing with semantic correlation maximization. In *AAAI*, pages 2177–2183, 2014.

[Zhang *et al.*, 2016] Hanwang Zhang, Meng Wang, Richang Hong, and Tat-Seng Chua. Play and rewind: Optimizing binary representations of videos by self-supervised temporal hashing. In *MM*, pages 781–790, 2016.