CHAIN: Exploring Global-Local Spatio-Temporal Information for Improved Self-Supervised Video Hashing

Rukai Wei
Huazhong University of Science and Technology
Wuhan, China
weirukai@hust.edu.cn

Yu Liu*
Huazhong University of Science and Technology
Wuhan, China
liu_yu@hust.edu.cn

Jingkuan Song
University of Electronic Science and Technology of China
Chengdu, China
jingkuan.song@gmail.com

Heng Cui
Huazhong University of Science and Technology
Wuhan, China
hengcui@hust.edu.cn

Yanzhao Xie
Huazhong University of Science and Technology
Wuhan, China
yzxie@hust.edu.cn

Ke Zhou
Huazhong University of Science and Technology
Wuhan, China
zhke@hust.edu.cn

ABSTRACT

Compressing videos into binary codes can improve retrieval speed and reduce storage overhead. However, learning accurate hash codes for video retrieval can be challenging due to high local redundancy and complex global dependencies between video frames, especially in the absence of labels. Existing self-supervised video hashing methods have been effective in designing expressive temporal encoders, but have not fully utilized the temporal dynamics and spatial appearance of videos due to less challenging and unreliable learning tasks. To address these challenges, we begin by utilizing the contrastive learning task to capture global spatio-temporal information of videos for hashing. With the aid of our designed augmentation strategies, which focus on spatial and temporal variations to create positive pairs, the learning framework can generate hash codes that are invariant to motion, scale, and viewpoint. Furthermore, we incorporate two collaborative learning tasks, *i.e.*, frame order verification and scene change regularization, to capture local spatio-temporal details within video frames, thereby enhancing the perception of temporal structure and the modeling of spatio-temporal relationships. Our proposed **C**ontrastive **H**ashing with Global-Local Sp**a**tio-temporal **I**nformatio**n** (CHAIN) outperforms state-of-the-art self-supervised video hashing methods on four video benchmark datasets. Our codes will be released.

CCS CONCEPTS

• **Computing methodologies** → **Visual content-based indexing and retrieval**; • **Information systems** → **Top-k retrieval in databases**.

*Corresponding author.

MM '23, October 29-November 3, 2023, Ottawa, ON, Canada

ACM ISBN 979-8-4007-0108-5/23/10…$15.00
https://doi.org/10.1145/3581783.3613440

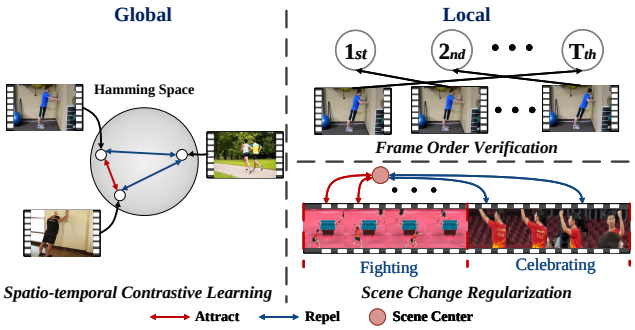


Figure 1: Illustration of the proposed three complementary learning tasks. Spatio-temporal contrastive learning concentrates on the global inter-video relationships, while frame order verification and scene change regularization focus on the local intra-video spatio-temporal details.

KEYWORDS

Self-supervised video hashing; Spatio-temporal contrastive learning; Frame order verification; Scene change regularization

ACM Reference Format:
Rukai Wei, Yu Liu, Jingkuan Song, Heng Cui, Yanzhao Xie, and Ke Zhou. 2023. CHAIN: Exploring Global-Local Spatio-Temporal Information for Improved Self-Supervised Video Hashing. In *Proceedings of the 31st ACM International Conference on Multimedia (MM '23), October 29-November 3, 2023, Ottawa, ON, Canada.* ACM, New York, NY, USA, 12 pages. https://doi.org/10.1145/3581783.3613440

1 INTRODUCTION

Videos, as common multimedia data, have recently received significant attention from researchers. As the demand for large-scale video retrieval increases in various multimedia applications, efficient retrieval and low storage costs have become urgent needs. To cope with these requirements, hashing-based methods [6, 32, 51, 60, 64] have emerged as the dominant solution for scalable video retrieval. They compress high-dimensional features into binary codes and preserve the similarities between samples in Hamming space, offering an efficient way to retrieve videos at scale. Since manual annotation

for large-scale video data is expensive and biased, self-supervised video hashing methods [15, 33, 34, 49] are in the spotlight.

Compared to 2D image retrieval, retrieving videos by self-supervised hashing is a challenging task due to the high local redundancy and complex global dependencies between video frames [9, 37]. To address this issue, existing self-supervised video hashing methods [15, 33, 34, 49] primarily focus on designing expressive learning models that capture temporal dependence. For instance, SSVH [49], BTH [34], and MCMSH [15] employ the LSTM model [19], the BERT model [8], and the MC-MLP model based on MLP-Mixer [52], respectively. Nevertheless, their learning tasks, such as *structure or neighborhood preserving*, focus on static spatial relationships, while neglecting valuable temporal cues, such as sequential order and changes over time, that are crucial for motion recognition. Therefore, it is imperative to develop more challenging learning tasks that can effectively capture both temporal and spatial information in videos, with the aim of enhancing retrieval performance.

To this end, we first design a contrastive learning task for self-supervised video hashing. To create high-quality positive pairs, we use spatial augmentations based on standard image contrastive learning methods [3, 14, 16], as well as temporal augmentations through segment sampling [25]. This spatio-temporal contrastive learning framework allows us to perceive both the spatial and temporal context of videos from unlabeled samples, and use the context as a supervisory signal to learn motion, scale, and viewpoint invariant hash codes. (See §3.2).

Note that the conventional contrastive learning task is primarily concerned with the global spatio-temporal representations of videos [10, 25, 58, 61]. As shown in the left segment of Figure 1, global information focuses on the correlation between video instances. However, local spatio-temporal details in the video frames are also critical to understanding the content of videos since they often provide key cues for distinguishing between different video instances. As shown in the right segment of Figure 1, the evolution of actions as well as variations in viewpoint within a video also provide crucial information for describing the video. To address this issue through learning tasks, we propose the frame order verification task and the scene change regularization task. For the former, we follow classic methods [28, 39] for predicting the absolute temporal position of frames in a sampled clip one by one (See §3.3), achieving full exploitation of the inherent sequential structure of a video. For the latter, we use intra-video prototypical contrastive learning [31], which enables the model to distinguish between various scenes within a video (See §3.4). This task can capture the necessary variation, which is crucial for robust spatio-temporal modeling. It can compensate for the weak perceptibility of transformer-based contrastive learning to frequent scene changes [61].

Finally, we integrate the above learning tasks into the contrastive learning framework and propose a novel method, namely, **C**ontrastive **H**ashing with Global-Local Sp**a**tio-Temporal **I**nformatio**n** (CHAIN). The spatio-temporal contrastive hashing task emphasizes the global relationships between video instances. Meanwhile, frame order verification captures the temporal connections between video frames, while scene change regulation focuses on spatial relationships between video frames. To accomplish these tasks effectively, the model is required to explore the multi-granularity context for robust spatio-temporal modeling. We conduct extensive experiments on four video benchmark datasets, including UCF-101 [50], HMDB-51 [26], FCVID [23], and ActivityNet [18]. Experimental results demonstrate that our proposed CHAIN outperforms state-of-the-art self-supervised video hashing methods by a large margin. Our main contributions can be outlined as follows:

- For global spatio-temporal relationships between video instances, we propose a novel spatio-temporal contrastive learning framework, where we define both spatial and temporal augmentations to create positive pairs, enhancing the model to learn motion, scale, and viewpoint invariant hash codes for videos.
- For local temporal connections between video frames, we introduce a frame order verification task to predict the absolute temporal positions of video frames, achieving full exploitation of the inherent sequential structure.
- For local spatial relationships between video frames, we incorporate a scene change regularization task to distinguish various scenes within a video, allowing the model to capture the spatio-temporal variations for robust spatio-temporal modeling.

2 RELATED WORK

In this paper, we focus on extending contrastive representation learning to self-supervised video hashing tasks. Therefore, our work is mainly related to the following two research areas:

Self-supervised video hashing. Previous research [13, 47, 57] considers frames within a video in isolation and learns hash codes without exploiting the temporal cues that naturally exist in videos, which fails to achieve satisfying retrieval performance. Subsequently, to model temporal relations between video frames, most recent works [63, 63] adopt RNN [62] and its variants LSTM [19] networks which are inherently suitable for capturing the sequence data, *e.g.*, SSVH [49], JATE [30], and NPH [33]. Most recently, BTH [34] uses a bidirectional transformer to mine more reliable temporal dependencies. MCMSH [15] proposes MC-MLP to separately model the three granularities of dependence. In addition, most methods define multi-tasks to learn hash codes in label-free scenarios. For example, SSVH, BTH, and MCMSH use similarity-preserving algorithms to reconstruct pairwise similarities between high-dimensional features in Hamming space. Meanwhile, they perform auxiliary tasks (*i.e.*, masked frames prediction, cluster alignment, inter/intra-class variation *etc.*) for a better understanding of the temporal structure.

Contrastive self-supervised learning. Contrastive learning is a popular technique for learning view-invariant representations by attracting positive samples and repelling negative ones. This technique has shown promising performance in image representation learning, with frameworks such as MoCo [16], SimCLR [3], BYOL [14], and SimSiam [4] being widely adopted in many downstream tasks. Recently, contrastive learning has been extended to video understanding. For example, VideoMoco [40] extends the image-based MoCo framework to video pre-training by modeling the temporal decay of the keys. In addition, SVT [45], CVRL [42], and VCLR [25] are established on the contrastive learning framework by defining various positive and negative sampling strategies.

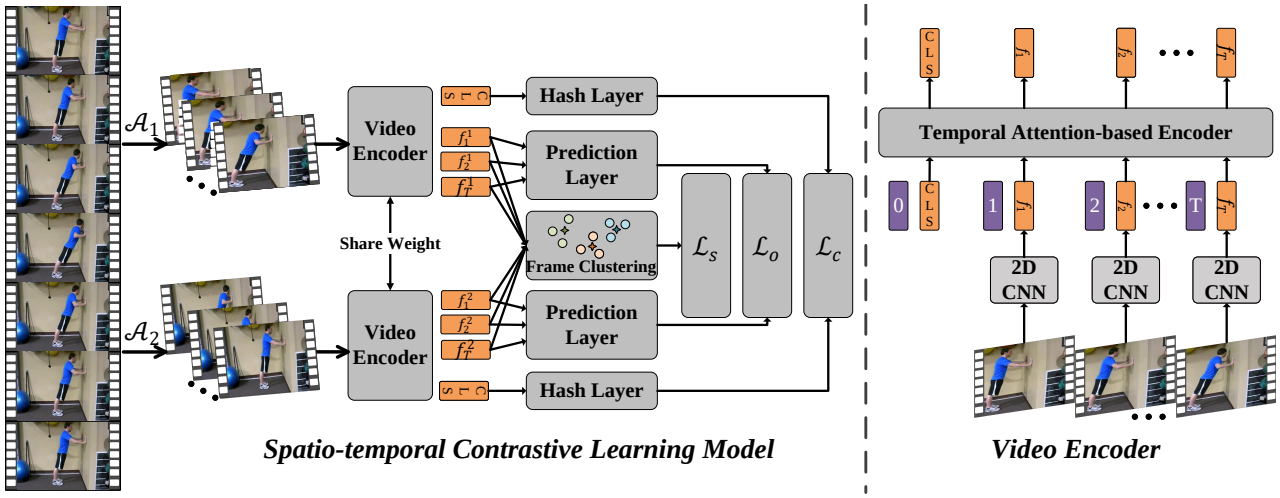


Figure 2: The overall framework of the proposed CHAIN. First, the proposed spatio-temporal augmentation strategies ($\mathcal{A}_1$ and $\mathcal{A}_2$) are applied to the input video, resulting in two augmented clips. Next, frame-level features are extracted using a 2D CNN, while long-range dependencies are modeled using a temporal attention-based encoder. Finally, a spatio-temporal contrastive learning framework is employed to learn high-quality hash codes. To better utilize spatio-temporal cues, two collaborative tasks are introduced in addition to the contrastive learning task. $\mathcal{L}_c$, $\mathcal{L}_o$, and $\mathcal{L}_s$ symbolize the objective functions of three learning tasks, respectively.

Improvements over existing jobs. Our proposed CHAIN leverages state-of-the-art video hashing methods for its model design, while utilizing spatio-temporal cues to learn video hash codes within a contrastive learning framework, resulting in an enhanced understanding of global video content. In addition, our frame order verification and scene change regulation tasks improve the model's ability to capture local spatio-temporal details. With the ability to fully exploit multi-granularity spatio-temporal information, our method achieves superior video hash learning.

3 METHOD

3.1 Problem Define and Overview

Given a training set $\mathcal{X} = \{x_i\}_{i=1}^{N}$ with N videos, we represent a video by its clip consisting of T frames, *i.e.*, $x_i \in \mathcal{R}^{T\times C\times H\times W}$, where C is the channel dimension and $H \times W$ is the spatial size. CHAIN aims to learn a nonlinear hash function that encodes each input x_i into a compact K-bit binary hash code b_i in the Hamming space, denoted as $x_i \rightarrow \{-1, 1\}^K$. Note that CHAIN does not require any supervision information from hand-crafted labels. The framework of CHAIN is shown in Figure 2.

Hash feature encoder. To process a sampled video clip $x = \{v_j\}_{j=1}^{T}$, we adopt the video encoder presented in Figure 2, which consists of a 2D CNN network and a attention-based transformer encoder. We first apply a 2D spatial encoder (*e.g.*, VGG [46] or ResNet50 [17] for fair comparisons with SOTA methods [15, 34, 49]) to extract the feature of each frame. Then, following BERT [8] and ViT [11], we add a special classification token (*i.e.*, $[CLS]$) in the front of the frame-level feature sequence. All these tokens will be added with position embeddings and then fed to the temporal attention-based encoder [54], where the attention mechanism will help to model the long-range dependencies. After that, we can obtain temporal-aware representations $\{f_j\}_{j=1}^{T} \cup \{z\}$, where z denotes the representation of x captured by the $[CLS]$ token, and f_j is the output embedding of frame v_j. For the global representation z, CHAIN will further project it into a real-valued hash code $h \in (-1, 1)^K$ with the hash layer, which contains two fully-connected layers. Finally, we discretize it into a binary code $b \in \{-1, 1\}^K$ by sgn.

Hash learning process. CHAIN learns high-quality hash codes under the contrastive learning framework. Given x_i, we first perform both spatial and temporal augmentations (See § 3.2) to generate two views, *i.e.*, $x_i^1 = \left\{v_j^1 \mid j = 1, ..., T\right\}$ and $x_i^2 = \left\{v_j^2 \mid j = 1, ..., T\right\}$. Note that these video clips may contain different video frames due to the randomness of temporal augmentation. Using the video encoder aforementioned, we can obtain the hash codes z_i^1 and z_i^2 as well as frame-level representations $f_{i,j}^1$ and $f_{i,j}^2$, where $j = 1, ..., T$. For the hash codes learning through global information, we adopt a standard contrastive learning objective to maximize the consistency between different views from the same video (See § 3.2). Regarding the frame-level representations, on one hand, we send them to the order prediction layer to infer their absolute positions in the video clip frame by frame (See § 3.3). On the other hand, all $f_{i,j}^1$ and $f_{i,j}^2$ will be clustered into several classes with the Affinity-Propagation algorithm [12] for scene change regularization (See § 3.4). By introducing three complementary learning tasks, CHAIN allows for a better comprehension of spatial and temporal cues, resulting in more robust hash codes for videos.

3.2 Spatio-temporal Contrastive Learning

In contrast to SOTA methods [15, 33, 49] that define a similarity preserving task to learn hash codes without the perception of temporal dynamics, we design a spatio-temporal contrastive learning

framework for superior video understanding. The instance-wise contrastive learning frameworks [3, 14, 16] aim to maximize the agreement between views that are augmented from the same instance. These frameworks rely on augmentation strategies to provide variance while preserving consistency, especially when applied to video learning. To this end, CHAIN applies both spatial and temporal augmentation to form high-quality positive pairs for videos.

Segment sampling based temporal augmentation. Temporal augmentation encourages the introduction of variance along the temporal dimension without altering video content. However, videos are more complicated than images with 2D spatial features due to the additional temporal dimension, which requires more effective augmentation strategies. Videos have two prominent characteristics: 1) Near frames are often redundant. 2) Videos depend on global context to tell the evolution of events. As a result, we follow the classic method [25] to perform segment-based frame sampling to learn from the global context. Specifically, for a video containing L frames, we divide it into T segments with equal duration. We randomly select one frame within each segment, and this allows us to obtain T frames in total that represent the video. By performing the random segment-level sampling twice, we can obtain two clips, *i.e.*, $\{v_1^1, v_2^1, ..., v_T^1\}$ and $\{v_1^2, v_2^2, ..., v_T^2\}$. They form a positive pair. Segment-level sampling has the potential to reduce local redundancy and provide a global view of the video for a better understanding of the entire event. Moreover, it allows the model to sample more scenes in the video, which is the foundation for implementing scene change regulation.

Temporally consistent spatial augmentation. Intuitively, we can apply existing image-based spatial augmentation methods, *e.g.*, RandomResizedCrop, RandomGrayScale, and RandomColorJitter, to videos frame-by-frame. However, since the randomness in cropping and blurring will break the continuity between frames, we follow the classic method [42] to fix the randomness across frames to make the spatial augmentation consistent along the temporal dimension. Note that spatial augmentation should differ across clips as our goal is to introduce variance between views. For example, we set the same random seed for all frames within a clip, while the seed for the other clip will be different.

Contrastive learning objective. We define the clips sampled from the same video as *positive pairs*, while the clips from other videos within a mini-batch are *negative pairs*. By attracting positive pairs and repelling negative pairs, contrastive learning enables the model to generate hash codes without annotations. Following classic methods [3, 38, 43], we can define the contrastive loss as follows:

$$\hat{\ell}_i^1 = -\log \frac{exp\left(sim(b_i^1, b_i^2)/\tau\right)}{exp\left(sim(b_i^1, b_i^2)/\tau\right) + \sum_{n,j\neq i} exp\left(sim(b_i^1, b_j^n)/\tau\right)}, \quad (1)$$

where ℓ_i^n is the loss of the n-th ($n = 1$ or 2) view of b_i, $sim(\cdot)$ can be defined as the cosine similarity, *i.e.*, $sim(b_i, b_j) = \frac{b_i^T b_j}{\|b_i\|\|b_j\|}$, and τ is the temperature parameter [3]. Furthermore, the spatio-temporal contrastive loss of all samples within a batch is

$$\mathcal{L}_c = \frac{1}{2 \cdot B} \sum_{i=1}^{B} (\ell_i^1 + \ell_i^2). \quad (2)$$

Binary constraint. The *sgn* function has a derivative of 0 almost everywhere, which makes it seemingly incompatible with back-propagation. This, in turn, renders the optimization problem above NP-hard and intractable. Following the commonly adopted method [33, 34], we use the straight-through estimator [5] to solve the binary optimization problem.

3.3 Frame Order Verification

Our proposed spatio-temporal contrastive learning with transformer leverages the global information for hash learning but is easily biased to neglect local temporal details [61]. To compensate for this deficiency, we follow the classic method [61] to define a frame order prediction task to use the temporal coherence as a supplementary supervisory signal.

Different from predicting a relative order between frames or clips [28, 39], the transformer's ability to capture long-term dependencies allows us to maintain temporal information within the video, frame-by-frame. This means we can assign the correct frame order as a self-supervised label, predicting the absolute temporal order of video frames. Specifically, we first obtain the representations of two self-augmented clips yielded by the video encoder, *i.e.*, $\left\{f_{i,1}^1, f_{i,2}^1, ..., f_{i,T}^1\right\}$ and $\left\{f_{i,1}^2, f_{i,2}^2, ..., f_{i,T}^2\right\}$, where $f_{i,j}^n$ is the representation of the j-th frame of the n-th view of x_i. Then, we train an auxiliary order prediction layer g_ξ (*i.e.*, a fully-connected layer followed by a softmax classifier parameterized by ξ) to reason the absolute position of each $f_{i,j}^n$ in the clip. We define the temporal order $y_j^{order} = j$ as the ground-truth label of the j-th frame. Therefore, the order verification loss can be described as a classification loss, *i.e.*,

$$\tilde{\ell}_i^n = \frac{1}{T} \sum_{j=1}^{T} CE\left(g_\xi(f_{i,j}^n), y_j^{order}\right), \quad (3)$$

$$\mathcal{L}_o = \frac{1}{2 \cdot B} \sum_{i=1}^{B} (\tilde{\ell}_i^1 + \tilde{\ell}_i^2), \quad (4)$$

where $\tilde{l}_i^n$ is the loss of the n-th clip with T frames augmented from x_i, and $CE(x, y)$ denotes the standard cross-entropy loss between an input x and a given label y, respectively.

3.4 Scene Change Regularization

Scene changes are ubiquitous in videos, especially in lengthy ones. As shown in Figure 1, scene changes bring a large shift in both the spatial background and the temporal motion. However, the attention-based transformer encoder tends to smooth temporal differences, weakening the discriminatory power of the contrastive model. Therefore, we propose a task for regulating scene changes that attracts similar frames while repelling frames that differ significantly from each other. We achieve this by leveraging prototypical contrastive learning [31] to regulate the model's ability to perceive spatio-temporal changes.

We collect all the frame-level representations $\left\{f_{i,1}^1, f_{i,2}^1, ..., f_{i,T}^1\right\}$ and $\left\{f_{i,1}^2, f_{i,2}^2, ..., f_{i,T}^2\right\}$ of x_i from the output of the transformer-based temporal encoder. Then, all these representations within a video will be clustered by the Affinity Propagation algorithm [12], where each

video can be adaptively divided into multiple scenes without the need to specify the number of scenes in advance. Intuitively, every cluster can be described as a common scene, where the prototype c_i represents the semantic center of this scene. Therefore, CHAIN contrasts between frame-level representations and scene prototypes with prototypical contrastive learning [31] to encourage frames within the same scene to be similarly represented while frames from different scenes are exactly the opposite. The prototypical contrastive loss is described below.

$$\bar{\ell}_{i,j}^{n} = -\log \frac{exp\left(sim(f_{i,j}^{n}, C(f_{i,j}^{n}))/\tau\right)}{exp\left(sim(f_{i,j}^{n}, C(f_{i,j}^{n}))/\tau\right) + \sum\limits_{c_k \neq C(f_{i,j}^{n})} exp\left(sim(f_{i,j}^{n}, c_k)/\tau\right)}, \quad (5)$$

where ℓ_i^n is the prototypical contrastive loss of the n-th ($n = 1$ or 2) view of f_i, $C(f_{i,j}^n)$ is the prototype that corresponds to $f_{i,j}^n$. The scene change regularization loss is defined below.

$$\mathcal{L}_s = \frac{1}{2 \cdot B \cdot T} \sum_{i=1}^{B} \sum_{j=1}^{T} \left(\bar{\ell}_{i,j}^{1} + \bar{\ell}_{i,j}^{2}\right). \quad (6)$$

Overall learning objective. Combining the three tasks, the overall learning objective can be derived as follows:

$$\mathcal{L} = \mathcal{L}_c + \mathcal{L}_o + \mathcal{L}_s, \quad (7)$$

which will jointly optimize the model by fully exploiting the multi-granularity Spatio-temporal context.

4 EXPERIMENTS

In this section, we conduct extensive experiments on four public benchmarks to demonstrate the superiority of our proposed CHAIN compared to state-of-the-art self-supervised video hashing methods. We also provide ablation studies and visualizations to verify the rationality of the design. More interesting experimental exploration can be found in the **supplementary material**.

4.1 Dataset and Evaluation Protocols

The datasets include UCF101 [50], HMDB51 [26], FCVID [23], ActivityNet [18]. In addition, we advocate the settings of [15, 34, 49, 63] to split the datasets for a fair comparison.

FCVID consists of 91,223 videos from 239 annotated categories. We follow [33, 34, 49] to employ 91,185 videos, of which 45,585 are used for training and the other 45,600 for evaluation.

UCF101 contains 13,320 videos spread across 101 categories of human actions. Since both the UCF101 and HMDB51 datasets provide official train/test splits, following classic methods [35, 60], we adopt the default setting of using 9,537 videos for training and retrieval and 3,783 videos from the test split as the query set.

HMDB51 contains 6,849 videos spread across 51 action categories. Referring to [35, 60], we use 3,570 videos for training and retrieval, and 1,530 videos from the test split as the query set.

ActivityNet v1.3 covers a broad range of human activities with 200 categories and a total of 14,950 videos, including 10,024 training videos and 4,926 validation videos. Due to some missing and damaged videos, we follow [15, 34] to use 9,992 videos as the training set and exploit the validation set as our test set. In addition, we randomly select 1,000 videos for queries and 3,919 videos as the retrieval set.

Evaluation protocols Following the same evaluation protocol as [15, 33, 34, 55], we utilize the mean Average Precision at top-K retrieved results (mAP@K) to evaluate the retrieval performance, where videos are sorted according to their Hamming distance and similar ones are required to be ranked higher in the retrieved result. Besides, we use Precision-Recall (PR) curves as an additional evaluation metric to provide a more comprehensive evaluation of the retrieval performance by considering the trade-off between precision and recall. We evaluate the retrieval results with code lengths of 16, 32, and 64 bits, following most baselines.

4.2 Implementation Details

Feature preparation To ensure fair comparisons, we follow [15, 33, 34] to sample 25 (*i.e.*, $T = 25$) frames for each video from UCF101, HMDB51, and FCVID datasets. We use VGG-16 [46] as the 2D CNN encoder to extract 4096-D frame-level features. For the ActivityNet dataset, we follow [15, 34] to sample 30 (*i.e.*, $T = 30$) frames per video and use ResNet-50 [17] to extract 2048-D frame-level features. Both VGG-16 and ResNet-50 are pre-trained on ImageNet [7], and features are pre-extracted before the training phase.

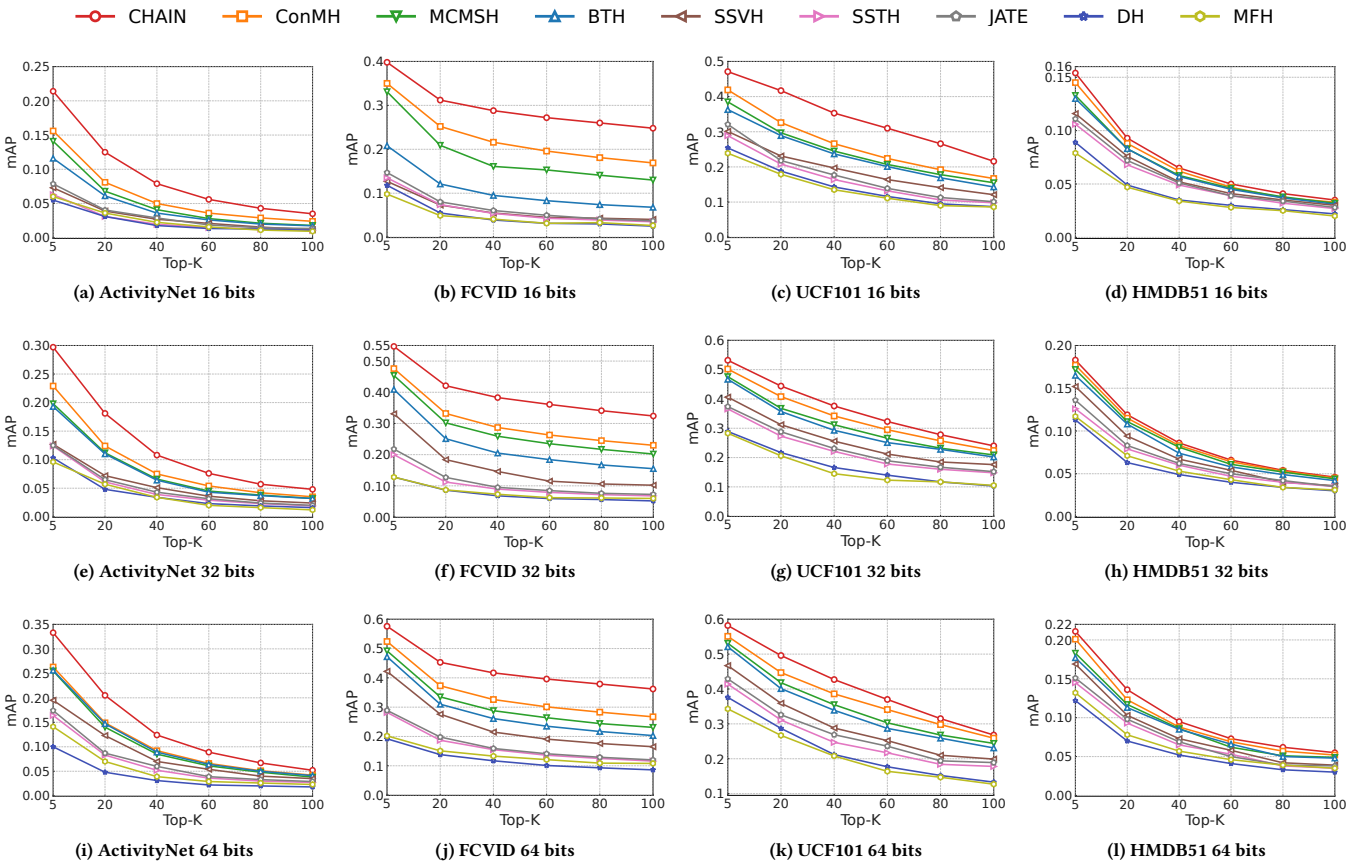
Model architecture. We use a single-layer transformer with a single attention head as the temporal attention-based encoder in our base model, and we also provide experimental results using different encoders, *e.g.*, MC-MLP [15]. Besides, we use a two-layer MLP, whose output dimension is K, as the hash layer. For the frame order prediction layer, we adopt single-layer MLP to predict the absolute position.

Training details. We set the batch size $B = 128$ and the initial learning rate as 0.0001, which will be decayed to 90% every 20 epochs with a minimal learning rate of 0.00001, following [55]. We optimize our model by using the Adam optimizer algorithm [24] with momentum 0.9. We implement our method in Pytorch with a single NVIDIA RTX 3090 GPU.

4.3 Comparison with State-of-the-arts

We compare CHAIN with several state-of-the-art self-supervised video hashing methods, including MFH [48], DH [36], JTAE [30], SSTH [63], SSVH [49], BTH [34], MCMSH [15], and ConMH [55]. For the results of baseline methods, we refer to ConMH [55].

mAP comparisons. As shown in Figure 3, CHAIN outperforms all the compared hashing methods, yielding remarkable results at all code lengths. Specifically, on the FCVID, UCF101, ActivityNet, and HMDB51 datasets, the mAP@20 values of CHAIN are 21.4%, 10.9%, 37.5%, and 10.5% higher than ConMH, respectively, with 64-bit hash codes. Note that CHAIN significantly outperforms ConMH, even with low-bit hash codes, *e.g.*, up to 23.8%, 27.9%, 54.3%, and 5.6% performance improvements on the four datasets at 16 bits, demonstrating its superiority in scenarios requiring high real-time performance and low storage demand. We attribute this advantage to the collaborative learning tasks, in which the contrastive learning task assists in capturing the video semantics by combining motion and stillness from a global perspective, and the order verification



(a) ActivityNet 16 bits (b) FCVID 16 bits (c) UCF101 16 bits (d) HMDB51 16 bits

(e) ActivityNet 32 bits (f) FCVID 32 bits (g) UCF101 32 bits (h) HMDB51 32 bits

(i) ActivityNet 64 bits (j) FCVID 64 bits (k) UCF101 64 bits (l) HMDB51 64 bits

Figure 3: Retrieval performance compared with state-of-the-art methods in terms of mAP@K over four datasets.

task and the scene change regularization task effectively mine the local temporal details.

We also observe that the improvement on HMDB51 is smaller than that on the other three datasets. We attribute this to two reasons. Firstly, the limited number of samples per class in HMDB51 is insufficient to fully stimulate the potential of the model. Secondly, inherent difficulties in motion recognition, such as shorter video duration, greater variability in camera viewpoints, and more complex interactions, make HMDB51 challenging. These challenges motivate us to explore more effective solutions in the future.

PR curves comparison. The precision-recall curves of BTH, MCMSH, ConMH, and CHAIN on FCVID and ActivityNet are shown in Figure 4. It is clear that CHAIN achieves higher precision at the same recall rate than other methods. These results demonstrate that CHAIN can achieve stable and superior retrieval performance. In particular, when a relatively low recall rate is acceptable, the precision advantage of CHAIN becomes even more prominent.

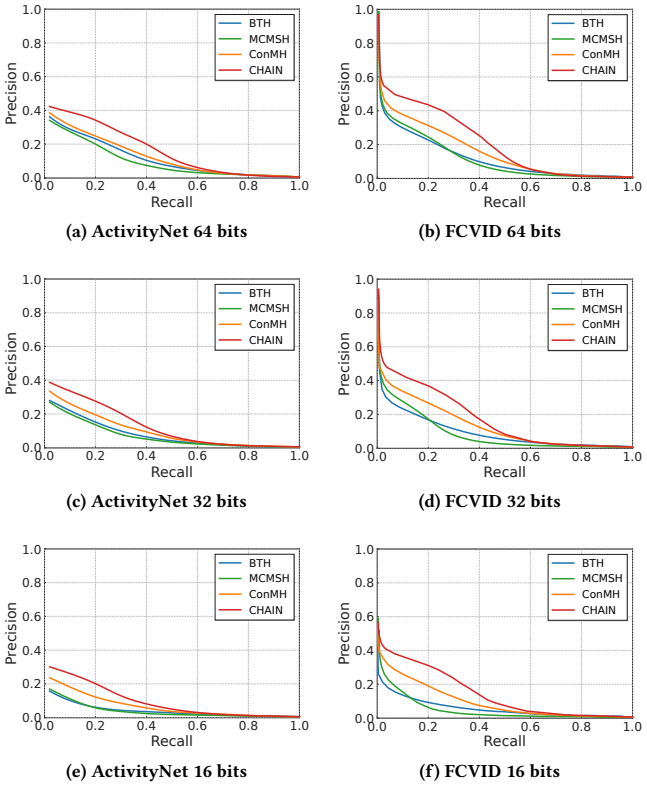
Cross-dataset performance comparison. To investigate the generalization of our CHAIN, we evaluate cross-dataset retrieval performance among different methods in this subsection. In detail, we train various methods on FCVID and test them on UCF101 and HMDB51. We compare the retrieval results with a single-dataset case, *i.e.*, training and testing on the same dataset. The mAP@20 results under the cross-dataset setting are reported in Table 1. All methods degrade in retrieval performance due to the domain gap between training and test data. Note that CHAIN still surpasses all baseline methods, and the drop in mAP (37% ↓ and 14% ↓) is relatively smaller than others. We credit this excellent generalization ability to the spatio-temporal contrastive learning framework. It ensures the alignment of positive samples, the divergence of class centers, and concentration of augmented data, which have been identified as three key factors contributing to the generalization ability according to recent research [21].

Table 1: Cross-dataset mAP@20 retrieval results and corresponding performance degrades of various methods, which are trained on FCVID and tested on UCF101 and HMDB51.

Method	BTH	MCMSH	ConMH	CHAIN
UCF101	0.244 (↓ 39%)	0.252 (↓ 40%)	0.278 (↓ 42%)	**0.309** (↓ 37%)
HMDB51	0.090 (↓ 17%)	0.091 (↓ 18%)	0.099 (↓ 19%)	**0.117** (↓ 14%)

4.4 Ablation Study

Effect of spatio-temporal augmentations. In CHAIN, we introduce both spatial and temporal augmentations to collaboratively



(a) ActivityNet 64 bits (b) FCVID 64 bits

(c) ActivityNet 32 bits (d) FCVID 32 bits

(e) ActivityNet 16 bits (f) FCVID 16 bits

Figure 4: PR curves of the proposed CHAIN and the state-of-the-art baselines, including ConMH, MCMSH, and BTH on FCVID and ActivityNet.

generate high-quality positive pairs for the proposed contrastive learning framework. We experiment on the augmentations to verify their effectiveness, and the results are reported in Table 2. We observed that contrastive learning with only spatial or temporal augmentation will result in sub-optimal performance (the first two rows vs. the third row). As a result, we conclude that both spatial and temporal information is essential for robust hash learning, and the proposed spatio-temporal augmentation strategy assists the model in focusing on the relevant spatio-temporal features.

Effect of frame order verification. We study the effect of the frame order verification (FOV) task in Table 2. Based on the comparison between the fourth row and the third row, we see an average improvement of 5.3% and 11.7% in UCF101 and ActivityNet, respectively. We believe that the FOV task provides a complementary signal to the contrastive learning task, enabling the model to better capture the temporal relationship between frames. These findings suggest that the FOV task is a valuable complement to contrastive learning in the context of spatio-temporal hashing.

Effect of scene change regularization. We also test the effect of scene change regulation (SCR) in Table 2. As seen a comparison between the fifth and third rows, scene change regulation focuses on frame-to-frame relationships and is an excellent complement to contrastive learning that focuses on video-to-video relationships. In addition, we find that the average improvement yielded by SCR

Table 2: The mAP@20 under different learning tasks. "CL" means contrastive learning, and "SA" and "TA" are spatial augmentation and temporal augmentation, respectively. Similarly, "FOV" and "SCR" stand for frame order verification and scene change regularization, respectively.

CL		FOV	SCR	UCF01			ActivityNet		
SA	TA			16bits	32bits	64bits	16bits	32bits	64bits
✓	✗	✗	✗	0.351	0.389	0.440	0.079	0.119	0.142
✗	✓	✗	✗	0.324	0.370	0.414	0.067	0.104	0.131
✓	✓	✗	✗	0.374	0.412	0.456	0.102	0.155	0.168
✓	✓	✓	✗	0.392	0.434	0.482	0.110	0.163	0.189
✓	✓	✗	✓	0.389	0.432	0.483	0.115	0.169	0.195
✓	✓	✓	✓	**0.417**	**0.444**	**0.496**	**0.125**	**0.181**	**0.205**

Table 3: The mAP@20 with different spatio-temporal augmentation strategies. Note that TCA denotes temporally consistent spatial augmentation, while w/o means without TCA, *i.e.*, the randomness in spatial augmentations are not fixed.

Augmentations	UCF101			ActivityNet		
	16bits	32bits	64bits	16bits	32bits	64bits
w/o TCA	0.414	0.438	0.491	0.122	0.174	0.197
w TCA	**0.417**	**0.444**	**0.496**	**0.125**	**0.181**	**0.205**
ramdom	0.368	0.423	0.467	0.108	0.157	0.178
consecutive	0.305	0.391	0.435	0.084	0.123	0.151
segment	**0.417**	**0.444**	**0.496**	**0.125**	**0.181**	**0.205**

on ActivityNet is larger than that on UCF101. We attribute it to the fact that the ActivityNet dataset contains videos captured from the internet. These videos include a diverse and rich set of real-world events and have longer duration compared to the videos in UCF101, resulting in a more prominent scene change problem.

Note that CHAIN cannot learn hash codes by FOV or SCR tasks solely, and we do not report corresponding results. They can only assist in model learning and do not directly impose constraints on the hash codes. If used alone, the hash layer lacks gradient information and cannot be updated, resulting in hash codes that lack semantic meaning and exhibit high randomness. Although these tasks do not directly optimize the hash layer, the gradients of these tasks can be back-propagated to the video encoder, capturing rich local details and enhancing spatio-temporal modeling for better input representations to the hash layer. As a result, our spatio-temporal contrastive learning framework incorporating both FOV and SCR tasks demonstrates a substantial improvement in retrieval performance, as shown in the last row of Table 2.

Investigation on different spatio-temporal augmentations. To determine the fit spatio-temporal augmentation strategy, we follow [42] to employ a temporally consistent spatial augmentation (TCA), *i.e.*, fix the randomness of spatial augmentation (*i.e.*, random color jitter) along temporal dimension. We evaluate hashing performance when configuring TCA in our approach. Based on a comparison between the first and second rows in Table 3, we decided to use TCA in the spatial augmentation process due to the slight performance improvements. In addition, we also test

Table 4: Encoding time of various deep hash methods.

Methods	BTH	MCMSH	ConMH	CHAIN
Encoding time	0.92ms	3.8ms	4.6ms	0.93ms

the influence of different temporal augmentation strategies in Table 3, including random sampling (*i.e.*, random sample T frames), consecutive sampling (*i.e.*, sample T consecutive frames), and our proposed segment-based sampling. In general, we find that the segment-based sampling strategy outperforms the others on all datasets. The use of segment-based sampling may be beneficial because it 1) covers a long range of a video, capturing more accurate global dependencies, and 2) acquires more scenes for scene change regulation, enhancing the model's discriminatory ability.

4.5 Encoding Time

The time it takes to generate binary codes is a crucial factor in practical retrieval systems [15, 34]. We count the encode time from frame-level features to the binary hash code, and record the average time for 100 videos in Table 4. Note that we run BTH, MCMSH, ConMH, and CHAIN on the same platform to ensure fairness. It is clear that CHAIN achieves significantly better results compared to BTH while requiring only a slightly longer encoding time (0.93ms vs 0.92ms). Furthermore, CHAIN outperforms the state-of-the-art method, ConMH, in terms of both retrieval performance and time complexity, where the time complexity of ConMH is measured using its small model consisting of 12 layers.

4.6 Qualitative Results

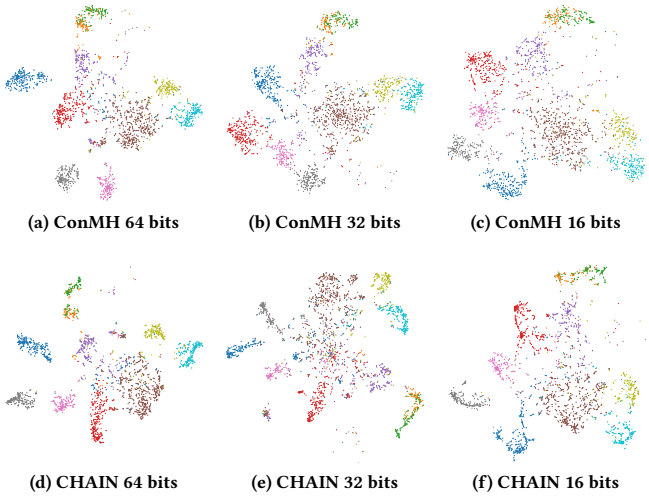
Top-5 retrieved results visualization. Figure 5 illustrates the top-5 retrieved results at 64 bits on FCVID, visually comparing the performance of CHAIN and ConMH. Following [15], we select four video classes, *i.e.*, "Hair cutting", "Making cookies", "Playing chess", and "Skiing", that exhibit complex human-object interactions and thus demand robust spatio-temporal modeling. Although both methods can provide relevant candidate videos, CHAIN exhibits a more stable performance in retrieving more relevant videos. For example, when given query videos from categories "Hair cutting" and "Making cookies", CHAIN achieves a top-5 precision of 80% and 40%, respectively, compared to ConMH's precision of 60% and 20%. This indicates that CHAIN is able to more effectively leverage the rich spatial and temporal information for hash learning.

The t-SNE visualization of hash codes. To better understand the results, we use t-SNE [53] to visualize the learned hash codes on the evaluation set of FCVID. As illustrated in Figure 6, we observed that the hash codes generated by CHAIN exhibit more distinct compactness for the same category and dispersion for dissimilar ones, in comparison to ConMH. This suggests that CHAIN can produce more discriminative binary codes, thereby offering remarkable retrieval performance.

5 CONCLUSION

In this paper, we propose CHAIN, a novel self-supervised video hashing algorithm that fully exploits both global and local spatio-temporal cues to achieve robust hash learning. On the one hand, we adopt a novel spatio-temporal contrastive learning framework to learn view-invariant hash codes. On the other hand, we incorporate two collaborative tasks, namely frame order verification and scene change regulation, to further enhance the model's spatio-temporal modeling capability. Experiments on four benchmarks demonstrate that our proposed CHAIN offers remarkable improvements compared with SOTA methods in terms of retrieval performance. CHAIN can be productively integrated into multimedia similarity search engines to support computation-efficient and storage-friendly video retrieval.

Figure 5: Top-5 retrieved results of CHAIN and ConMH on FCVID dataset. The video inside the green square is correctly retrieved, while the video inside the red square is incorrect.



(a) ConMH 64 bits (b) ConMH 32 bits (c) ConMH 16 bits

(d) CHAIN 64 bits (e) CHAIN 32 bits (f) CHAIN 16 bits

Figure 6: The t-SNE visualization of the learned 64-bit hash codes from the test set of FCVID. The scattered points of the same color indicate the same category. Note that we only visualize the first 10 classes.

ACKNOWLEDGMENTS

This work was achieved in Key Laboratory of Information Storage System and Ministry of Education of China. It was supported by the National Natural Science Foundation of China No.61902135, the National Natural Science Foundation of China (key program) No.62232007, and the Natural Science Foundation of Hubei Province No.2022CFB060.

REFERENCES

[1] Alberto Baldrati, Marco Bertini, Tiberio Uricchio, and Alberto Del Bimbo. 2022. Effective conditioned and composed image retrieval combining CLIP-based features. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2022, New Orleans, LA, USA, June 18-24, 2022*. IEEE, 21434–21442.

[2] Gedas Bertasius, Heng Wang, and Lorenzo Torresani. 2021. Is Space-Time Attention All You Need for Video Understanding?. In *Proceedings of the 38th International Conference on Machine Learning, ICML 2021, 18-24 July 2021, Virtual Event (Proceedings of Machine Learning Research, Vol. 139)*, Marina Meila and Tong Zhang (Eds.). PMLR, 813–824.

[3] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey E. Hinton. 2020. A Simple Framework for Contrastive Learning of Visual Representations. In *Proceedings of the 37th International Conference on Machine Learning, ICML 2020, 13-18 July 2020, Virtual Event (Proceedings of Machine Learning Research, Vol. 119)*. PMLR, 1597–1607.

[4] Xinlei Chen and Kaiming He. 2021. Exploring Simple Siamese Representation Learning. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2021, virtual, June 19-25, 2021*. Computer Vision Foundation / IEEE, 15750–15758.

[5] Matthieu Courbariaux, Yoshua Bengio, and Jean-Pierre David. 2015. BinaryConnect: Training Deep Neural Networks with binary weights during propagations. In *Advances in Neural Information Processing Systems 28: Annual Conference on Neural Information Processing Systems 2015, December 7-12, 2015, Montreal, Quebec, Canada*, Corinna Cortes, Neil D. Lawrence, Daniel D. Lee, Masashi Sugiyama, and Roman Garnett (Eds.). 3123–3131.

[6] Hui Cui, Lei Zhu, Jingjing Li, Zheng Zhang, and Weili Guan. 2022. Webly Supervised Image Hashing with Lightweight Semantic Transfer Network. In *Proceedings of the 30th ACM International Conference on Multimedia* (Lisboa, Portugal) *(MM '22)*. Association for Computing Machinery, New York, NY, USA, 3451–3460.

[7] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. 2009. ImageNet: A large-scale hierarchical image database. In *2009 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR 2009), 20-25 June 2009, Miami, Florida, USA*. IEEE Computer Society, 248–255.

[8] 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, NAACL-HLT 2019, Minneapolis, MN, USA, June 2-7, 2019, Volume 1 (Long and Short Papers)*. Association for Computational Linguistics, 4171–4186.

[9] Jianfeng Dong, Xianke Chen, Minsong Zhang, Xun Yang, Shujie Chen, Xirong Li, and Xun Wang. 2022. Partially Relevant Video Retrieval. In *Proceedings of the 30th ACM International Conference on Multimedia*. 246–257.

[10] Jianfeng Dong, Xirong Li, Chaoxi Xu, Xun Yang, Gang Yang, Xun Wang, and Meng Wang. 2022. Dual encoding for video retrieval by text. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 44, 8 (2022), 4065–4080.

[11] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. 2021. An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale. In *9th International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, May 3-7, 2021*. OpenReview.net.

[12] Brendan J Frey and Delbert Dueck. 2007. Clustering by passing messages between data points. *science* 315, 5814 (2007), 972–976.

[13] Yunchao Gong and Svetlana Lazebnik. 2011. Iterative quantization: A procrustean approach to learning binary codes. In *The 24th IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2011, Colorado Springs, CO, USA, 20-25 June 2011*. IEEE Computer Society, 817–824.

[14] Jean-Bastien Grill, Florian Strub, Florent Altché, Corentin Tallec, Pierre H. Richemond, Elena Buchatskaya, Carl Doersch, Bernardo Ávila Pires, Zhaohan Guo, Mohammad Gheshlaghi Azar, Bilal Piot, Koray Kavukcuoglu, Rémi Munos, and Michal Valko. 2020. Bootstrap Your Own Latent - A New Approach to Self-Supervised Learning. (2020).

[15] Yanbin Hao, Jingru Duan, Hao Zhang, Bin Zhu, Pengyuan Zhou, and Xiangnan He. 2022. Unsupervised Video Hashing with Multi-granularity Contextualization and Multi-structure Preservation. In *MM '22: The 30th ACM International Conference on Multimedia, Lisboa, Portugal, October 10 - 14, 2022*. ACM, 3754–3763.

[16] Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross B. Girshick. 2020. Momentum Contrast for Unsupervised Visual Representation Learning. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2020, Seattle, WA, USA, June 13-19, 2020*. Computer Vision Foundation / IEEE, 9726–9735.

[17] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2016. Deep Residual Learning for Image Recognition. In *2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2016, Las Vegas, NV, USA, June 27-30, 2016*. IEEE Computer Society, 770–778.

[18] Fabian Caba Heilbron, Victor Escorcia, Bernard Ghanem, and Juan Carlos Niebles. 2015. ActivityNet: A large-scale video benchmark for human activity understanding. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2015, Boston, MA, USA, June 7-12, 2015*. IEEE Computer Society, 961–970.

[19] Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long Short-Term Memory. *Neural Comput.* 9, 8 (1997), 1735–1780. https://doi.org/10.1162/neco.1997.9.8.1735

[20] Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long Short-Term Memory. *Neural Comput.* 9, 8 (1997), 1735–1780.

[21] Weiran Huang, Mingyang Yi, Xuyang Zhao, and Zihao Jiang. 2023. Towards the Generalization of Contrastive Self-Supervised Learning. In *The Eleventh International Conference on Learning Representations, ICLR 2023*.

[22] Menglin Jia, Luming Tang, Bor-Chun Chen, Claire Cardie, Serge J. Belongie, Bharath Hariharan, and Ser-Nam Lim. 2022. Visual Prompt Tuning. In *Computer Vision - ECCV 2022 - 17th European Conference, Tel Aviv, Israel, October 23-27, 2022, Proceedings, Part XXXIII (Lecture Notes in Computer Science, Vol. 13693)*, Shai Avidan, Gabriel J. Brostow, Moustapha Cissé, Giovanni Maria Farinella, and Tal Hassner (Eds.). Springer, 709–727.

[23] Yu-Gang Jiang, Zuxuan Wu, Jun Wang, Xiangyang Xue, and Shih-Fu Chang. 2018. Exploiting Feature and Class Relationships in Video Categorization with Regularized Deep Neural Networks. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 40, 2 (2018), 352–364.

[24] Diederik P. Kingma and Jimmy Ba. 2015. Adam: A Method for Stochastic Optimization. In *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*, Yoshua Bengio and Yann LeCun (Eds.).

[25] Haofei Kuang, Yi Zhu, Zhi Zhang, Xinyu Li, Joseph Tighe, Sören Schwertfeger, Cyrill Stachniss, and Mu Li. 2021. Video Contrastive Learning with Global Context. In *IEEE/CVF International Conference on Computer Vision Workshops, ICCVW 2021, Montreal, BC, Canada, October 11-17, 2021*. IEEE, 3188.

[26] Hildegard Kuehne, Hueihan Jhuang, Estíbaliz Garrote, Tomaso A. Poggio, and Thomas Serre. 2011. HMDB: A large video database for human motion recognition. In *IEEE International Conference on Computer Vision, ICCV 2011, Barcelona, Spain, November 6-13, 2011*, Dimitris N. Metaxas, Long Quan, Alberto Sanfeliu, and Luc Van Gool (Eds.). IEEE Computer Society, 2556–2563.

[27] Gihyun Kwon and Jong Chul Ye. 2022. CLIPstyler: Image Style Transfer with a Single Text Condition. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2022, New Orleans, LA, USA, June 18-24, 2022*. IEEE, 18041–18050.

[28] Hsin-Ying Lee, Jia-Bin Huang, Maneesh Singh, and Ming-Hsuan Yang. 2017. Unsupervised Representation Learning by Sorting Sequences. In *IEEE International Conference on Computer Vision, ICCV 2017, Venice, Italy, October 22-29, 2017*. IEEE Computer Society, 667–676.

[29] Seongyeong Lee, Hansoo Park, Dong Uk Kim, Jihyeon Kim, Muhammadjon Boboev, and Seungryul Baek. 2023. Image-free Domain Generalization via CLIP for 3D Hand Pose Estimation. In *IEEE/CVF Winter Conference on Applications of Computer Vision, WACV 2023, Waikoloa, HI, USA, January 2-7, 2023*. IEEE, 2933–2943.

[30] Chao Li, Yang Yang, Jiewei Cao, and Zi Huang. 2017. Jointly Modeling Static Visual Appearance and Temporal Pattern for Unsupervised Video Hashing. In *Proceedings of the 2017 ACM on Conference on Information and Knowledge Management, CIKM 2017, Singapore, November 06 - 10, 2017*. ACM, 9–17.

[31] Junnan Li, Pan Zhou, Caiming Xiong, and Steven Hoi. 2021. Prototypical Contrastive Learning of Unsupervised Representations. In *International Conference on Learning Representations, ICLR 2021*.

[32] Liang Li, Baihua Zheng, and Weiwei Sun. 2022. Adaptive Structural Similarity Preserving for Unsupervised Cross Modal Hashing. In *Proceedings of the 30th ACM International Conference on Multimedia* (Lisboa, Portugal) *(MM '22)*. Association for Computing Machinery, New York, NY, USA, 3712–3721.

[33] Shuyan Li, Zhixiang Chen, Jiwen Lu, Xiu Li, and Jie Zhou. 2019. Neighborhood Preserving Hashing for Scalable Video Retrieval. In *2019 IEEE/CVF International Conference on Computer Vision, ICCV 2019, Seoul, Korea (South), October 27 - November 2, 2019*. IEEE, 8211–8220.

[34] Shuyan Li, Xiu Li, Jiwen Lu, and Jie Zhou. 2021. Self-Supervised Video Hashing via Bidirectional Transformers. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2021, virtual, June 19-25, 2021*. Computer Vision Foundation / IEEE, 13549–13558.

[35] Yunqiang Li and Jan van Gemert. 2021. Deep Unsupervised Image Hashing by Maximizing Bit Entropy. In *Thirty-Fifth AAAI Conference on Artificial Intelligence, AAAI 2021, Thirty-Third Conference on Innovative Applications of Artificial Intelligence, IAAI 2021, The Eleventh Symposium on Educational Advances in Artificial Intelligence, EAAI 2021, Virtual Event, February 2-9, 2021*. AAAI Press, 2002–2010.

[36] Venice Erin Liong, Jiwen Lu, Gang Wang, Pierre Moulin, and Jie Zhou. 2015. Deep hashing for compact binary codes learning. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2015, Boston, MA, USA, June 7-12, 2015*. IEEE Computer Society, 2475–2483.

[37] Dezhao Luo, Jiabo Huang, Shaogang Gong, Hailin Jin, and Yang Liu. 2023. Towards Generalisable Video Moment Retrieval: Visual-Dynamic Injection to Image-Text Pre-Training. *CoRR* abs/2303.00040 (2023).

[38] Xiao Luo, Daqing Wu, Zeyu Ma, Chong Chen, Minghua Deng, Jianqiang Huang, and Xian-Sheng Hua. 2021. A Statistical Approach to Mining Semantic Similarity for Deep Unsupervised Hashing. In *MM '21: ACM Multimedia Conference, Virtual*

Event, China, October 20 - 24, 2021. ACM, 4306–4314.

[39] Ishan Misra, C. Lawrence Zitnick, and Martial Hebert. 2016. Shuffle and Learn: Unsupervised Learning Using Temporal Order Verification. In *Computer Vision - ECCV 2016 - 14th European Conference, Amsterdam, The Netherlands, October 11-14, 2016, Proceedings, Part I (Lecture Notes in Computer Science, Vol. 9905)*, Bastian Leibe, Jiri Matas, Nicu Sebe, and Max Welling (Eds.). Springer, 527–544.

[40] Tian Pan, Yibing Song, Tianyu Yang, Wenhao Jiang, and Wei Liu. 2021. VideoMoCo: Contrastive Video Representation Learning With Temporally Adversarial Examples. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2021, virtual, June 19-25, 2021*. Computer Vision Foundation / IEEE, 11205–11214.

[41] AJ Piergiovanni, Weicheng Kuo, and Anelia Angelova. 2022. Rethinking Video ViTs: Sparse Video Tubes for Joint Image and Video Learning. *CoRR* abs/2212.03229 (2022).

[42] Rui Qian, Tianjian Meng, Boqing Gong, Ming-Hsuan Yang, Huisheng Wang, Serge J. Belongie, and Yin Cui. 2021. Spatiotemporal Contrastive Video Representation Learning. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2021, virtual, June 19-25, 2021*. Computer Vision Foundation / IEEE, 6964–6974.

[43] Zexuan Qiu, Qinliang Su, Zijing Ou, Jianxing Yu, and Changyou Chen. 2021. Unsupervised Hashing with Contrastive Information Bottleneck. In *Proceedings of the Thirtieth International Joint Conference on Artificial Intelligence, IJCAI 2021, Virtual Event / Montreal, Canada, 19-27 August 2021*, Zhi-Hua Zhou (Ed.). ijcai.org, 959–965.

[44] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. 2021. Learning Transferable Visual Models From Natural Language Supervision. In *Proceedings of the 38th International Conference on Machine Learning, ICML 2021, 18-24 July 2021, Virtual Event (Proceedings of Machine Learning Research, Vol. 139)*, Marina Meila and Tong Zhang (Eds.). PMLR, 8748–8763.

[45] Kanchana Ranasinghe, Muzammal Naseer, Salman Khan, Fahad Shahbaz Khan, and Michael S. Ryoo. 2022. Self-supervised Video Transformer. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2022, New Orleans, LA, USA, June 18-24, 2022*. IEEE, 2864–2874.

[46] Karen Simonyan and Andrew Zisserman. 2015. Very Deep Convolutional Networks for Large-Scale Image Recognition. (2015).

[47] Jingkuan Song, Yi Yang, Zi Huang, Heng Tao Shen, and Richang Hong. 2011. Multiple Feature Hashing for Real-Time Large Scale near-Duplicate Video Retrieval. In *Proceedings of the 19th ACM International Conference on Multimedia* (Scottsdale, Arizona, USA) *(MM '11)*. Association for Computing Machinery, New York, NY, USA, 423–432. https://doi.org/10.1145/2072298.2072354

[48] Jingkuan Song, Yi Yang, Zi Huang, Heng Tao Shen, and Richang Hong. 2011. Multiple feature hashing for real-time large scale near-duplicate video retrieval. In *Proceedings of the 19th International Conference on Multimedia 2011, Scottsdale, AZ, USA, November 28 - December 1, 2011*. ACM, 423–432.

[49] Jingkuan Song, Hanwang Zhang, Xiangpeng Li, Lianli Gao, Meng Wang, and Richang Hong. 2018. Self-Supervised Video Hashing With Hierarchical Binary Auto-Encoder. *IEEE Trans. Image Process.* 27, 7 (2018), 3210–3221.

[50] Khurram Soomro, Amir Roshan Zamir, and Mubarak Shah. 2012. UCF101: A Dataset of 101 Human Actions Classes From Videos in The Wild. *CoRR* abs/1212.0402 (2012).

[51] Yuan Sun, Dezhong Peng, Haixiao Huang, and Zhenwen Ren. 2022. Feature and Semantic Views Consensus Hashing for Image Set Classification. In *Proceedings of the 30th ACM International Conference on Multimedia* (Lisboa, Portugal) *(MM '22)*. Association for Computing Machinery, New York, NY, USA, 2097–2105.

[52] Ilya O. Tolstikhin, Neil Houlsby, Alexander Kolesnikov, Lucas Beyer, Xiaohua Zhai, Thomas Unterthiner, Jessica Yung, Andreas Steiner, Daniel Keysers, Jakob Uszkoreit, Mario Lucic, and Alexey Dosovitskiy. 2021. MLP-Mixer: An all-MLP Architecture for Vision. In *Advances in Neural Information Processing Systems 34: Annual Conference on Neural Information Processing Systems 2021, NeurIPS 2021, December 6-14, 2021, virtual*. 24261–24272.

[53] Laurens van der Maaten and Geoffrey E. Hinton. 2008. Visualizing Data using t-SNE. *Journal of Machine Learning Research* 9 (2008), 2579–2605.

[54] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention is All you Need. In *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA*, Isabelle Guyon, Ulrike von Luxburg, Samy Bengio, Hanna M. Wallach, Rob Fergus, S. V. N. Vishwanathan, and Roman Garnett (Eds.). 5998–6008.

[55] Yuting Wang, Jinpeng Wang, Bin Chen, Ziyun Zeng, and Shu-Tao Xia. 2023. Contrastive Masked Autoencoders for Self-Supervised Video Hashing. In *Proceedings of the AAAI Conference on Artificial Intelligence*.

[56] Zhaoqing Wang, Yu Lu, Qiang Li, Xunqiang Tao, Yandong Guo, Mingming Gong, and Tongliang Liu. 2022. CRIS: CLIP-Driven Referring Image Segmentation. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2022, New Orleans, LA, USA, June 18-24, 2022*. IEEE, 11676–11685.

[57] Yair Weiss, Antonio Torralba, and Rob Fergus. 2008. Spectral Hashing. In *Proceedings of the 21st International Conference on Neural Information Processing Systems* (Vancouver, British Columbia, Canada) *(NIPS'08)*. Curran Associates Inc., Red Hook, NY, USA, 1753–1760.

[58] Ting Yao, Yiheng Zhang, Zhaofan Qiu, Yingwei Pan, and Tao Mei. 2021. SeCo: Exploring Sequence Supervision for Unsupervised Representation Learning. In *Thirty-Fifth AAAI Conference on Artificial Intelligence, AAAI 2021, Thirty-Third Conference on Innovative Applications of Artificial Intelligence, IAAI 2021, The Eleventh Symposium on Educational Advances in Artificial Intelligence, EAAI 2021, Virtual Event, February 2-9, 2021*. AAAI Press, 10656–10664.

[59] Yingchen Yu, Fangneng Zhan, Rongliang Wu, Jiahui Zhang, Shijian Lu, Miaomiao Cui, Xuansong Xie, Xian-Sheng Hua, and Chunyan Miao. 2022. Towards Counterfactual Image Manipulation via CLIP. In *MM '22: The 30th ACM International Conference on Multimedia, Lisboa, Portugal, October 10 - 14, 2022*, João Magalhães, Alberto Del Bimbo, Shin'ichi Satoh, Nicu Sebe, Xavier Alameda-Pineda, Qin Jin, Vincent Oria, and Laura Toni (Eds.). ACM, 3637–3645.

[60] Li Yuan, Tao Wang, Xiaopeng Zhang, Francis E. H. Tay, Zequn Jie, Wei Liu, and Jiashi Feng. 2020. Central Similarity Quantization for Efficient Image and Video Retrieval. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2020, Seattle, WA, USA, June 13-19, 2020*. Computer Vision Foundation / IEEE, 3080–3089.

[61] Sukmin Yun, Jaehyung Kim, Dongyoon Han, Hwanjun Song, Jung-Woo Ha, and Jinwoo Shin. 2022. Time Is MattEr: Temporal Self-supervision for Video Transformers. In *International Conference on Machine Learning, ICML 2022, 17-23 July 2022, Baltimore, Maryland, USA (Proceedings of Machine Learning Research, Vol. 162)*, Kamalika Chaudhuri, Stefanie Jegelka, Le Song, Csaba Szepesvári, Gang Niu, and Sivan Sabato (Eds.). PMLR, 25804–25816.

[62] Wojciech Zaremba, Ilya Sutskever, and Oriol Vinyals. 2014. Recurrent Neural Network Regularization. *CoRR* abs/1409.2329 (2014). arXiv:1409.2329

[63] Hanwang Zhang, Meng Wang, Richang Hong, and Tat-Seng Chua. 2016. Play and Rewind: Optimizing Binary Representations of Videos by Self-Supervised Temporal Hashing. In *Proceedings of the 2016 ACM Conference on Multimedia Conference, MM 2016, Amsterdam, The Netherlands, October 15-19, 2016*, Alan Hanjalic, Cees Snoek, Marcel Worring, Dick C. A. Bulterman, Benoit Huet, Aisling Kelliher, Yiannis Kompatsiaris, and Jin Li (Eds.). ACM, 781–790.

[64] Han Zhu, Mingsheng Long, Jianmin Wang, and Yue Cao. 2016. Deep Hashing Network for Efficient Similarity Retrieval. In *Proceedings of the Thirtieth AAAI Conference on Artificial Intelligence, February 12-17, 2016, Phoenix, Arizona, USA*, Dale Schuurmans and Michael P. Wellman (Eds.). AAAI Press, 2415–2421.

A RETRIEVAL PERFORMANCE WITH CLIP

Large-scale pre-trained models are known to be strong vision learners, and in particular, the CLIP model[44] has demonstrated remarkable capabilities in various tasks[1, 27, 29, 56, 59]. Compared to the CNN network used in our baseline model, the CLIP model has proven to excel in image feature extraction, owing to its pre-training on large-scale image-text datasets and experssive model design. To investigate how CHAIN can benefit from a stronger feature extraction backbone, we use the CLIP visual branch (*i.e.*, ViT-B/16 [11]) to replace the CNN backbones. We evaluate the performance of CHAIN in different datasets under such a stronger setting, and the results are shown in Figure 7.

We can observe that CLIP can provide significant performance improvements. For instance, on the ActivityNet and UCF101 datasets, CHAIN +CLIP achieves mAP@20 values that are 47.8% and 44.1% higher than the baseline models that use VGG16 [46] and ResNet50 [17], respectively, when using 64-bit hash codes. Even with low-bit hash codes, CLIP brings remarkable enhancements, improving performance by up to 91.2% and 51.5% on the two datasets at 16 bits.

The CLIP model has great potential in video hashing tasks, as demonstrated by our initial results. However, our current approach only utilizes the pre-trained vision branch as our 2D frame feature extractor, without fully exploring its capabilities, particularly in zero-shot scenarios. To address this, we aim to investigate how to leverage visual prompt [22] learning for video hashing and explore ways to incorporate the text branch to further enhance the model's understanding of videos.

B PERFORMANCE VARYING WITH NUMBER OF SAMPLED FRAMES

For fair comparison, we set the number of sampled frames $T = 25$ for UCF101 and $T = 30$ for ActivityNet in our experiments. We investigate the influence of T on retrieval performance, and results are illustrated in Figure 8. Our findings indicate that sampling 50 frames (*i.e.*, $T = 50$) does not offer significant performance improvements, and may even result in degradation in certain datasets, such as UCF101 at 64 bits. The reason for this is that the videos in these datasets are relatively short and contain relatively simple content, and sampling too many frames introduces unnecessary redundancy. However, we observed a slight decrease in retrieval performance when the number of frames was reduced ($T = 8$ v.s $T = 25$ or 30).

C PERFORMANCE WITH DIFFERENT TEMPORAL ENCODERS

Note that the temporal encoder in our method is implementation-agnostic, allowing the integration of the latest models in this area. As a result, we also investigate the retrieval performance of CHAIN when using a more distinct temporal encoder, such as MC-MLP [15]. In addition to the transformer encoder-based model mentioned above, we also evaluate other temporal encoders, namely LSTM [20] from SSTH [63] and MC-MLP from MCMSH [15]. The results are presented in Table 5. These results demonstrate that the adoption of MC-MLP and transformer encoders can significantly enhance the performance of CHAIN.

D EXPERIMENTAL ENVIRONMENT

The experiments are conducted on a machine with Intel(R) Xeon(R) Gold 6226R CPU @ 2.90GHz, and a single NVIDIA RTX 3090 GPU with 24GB GPU memory. The operating system of the machine is Ubuntu 20.04.5 LTS. For software versions, we use Python 3.9.16, Pytorch 1.12.1, and CUDA 11.7.

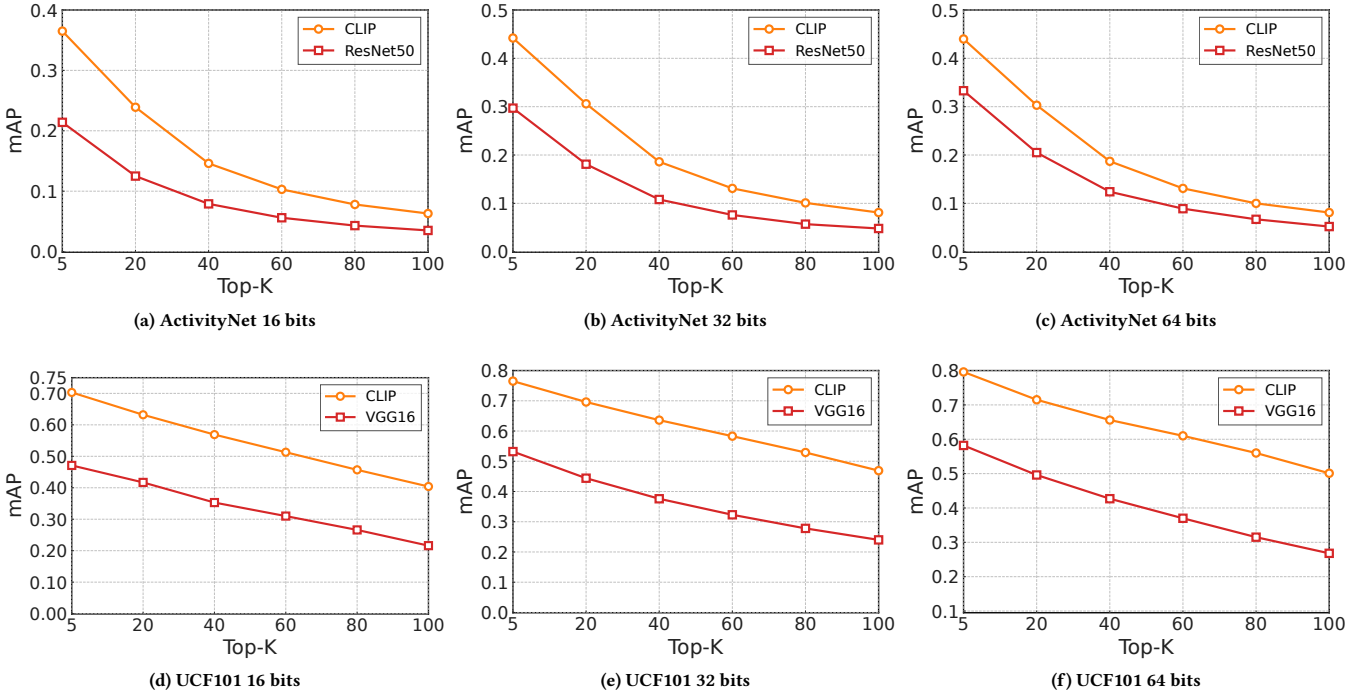
E LIMITATIONS AND FUTURE WORK

Although our proposed CHAIN achieves remarkable retrieval performance, there are still some limitations that need to be solved in our future work.

- **Under-exploration of more advanced backbones.** Although we use VGG/ResNet as the spatial encoder and a transformer encoder to model temporal dynamics for fair comparisons, it falls short in dealing with real-world long videos. We plan to extend our method to more recent backbones such as TimeSformer [2] and TubeViT [41].
- **High-cost in model deployment.** Although the time complexity of CHAIN is in line with the baselines, its deployment in real-time applications remains challenging. We aim to explore computationally efficient models and compress the current model to better meet the needs of scenarios with high real-time requirements.
- **Expensive model updating.** The over-time changes in the distribution of real-world video data will degrade the retrieval performance. To address these issues, we aim to explore more effective updating strategies that can reduce the cost of re-training the model and refreshing hashing indexes.

Table 5: Comparison of mAP@K using different temporal encoders on the ActivityNet dataset with 64-bit hash codes.

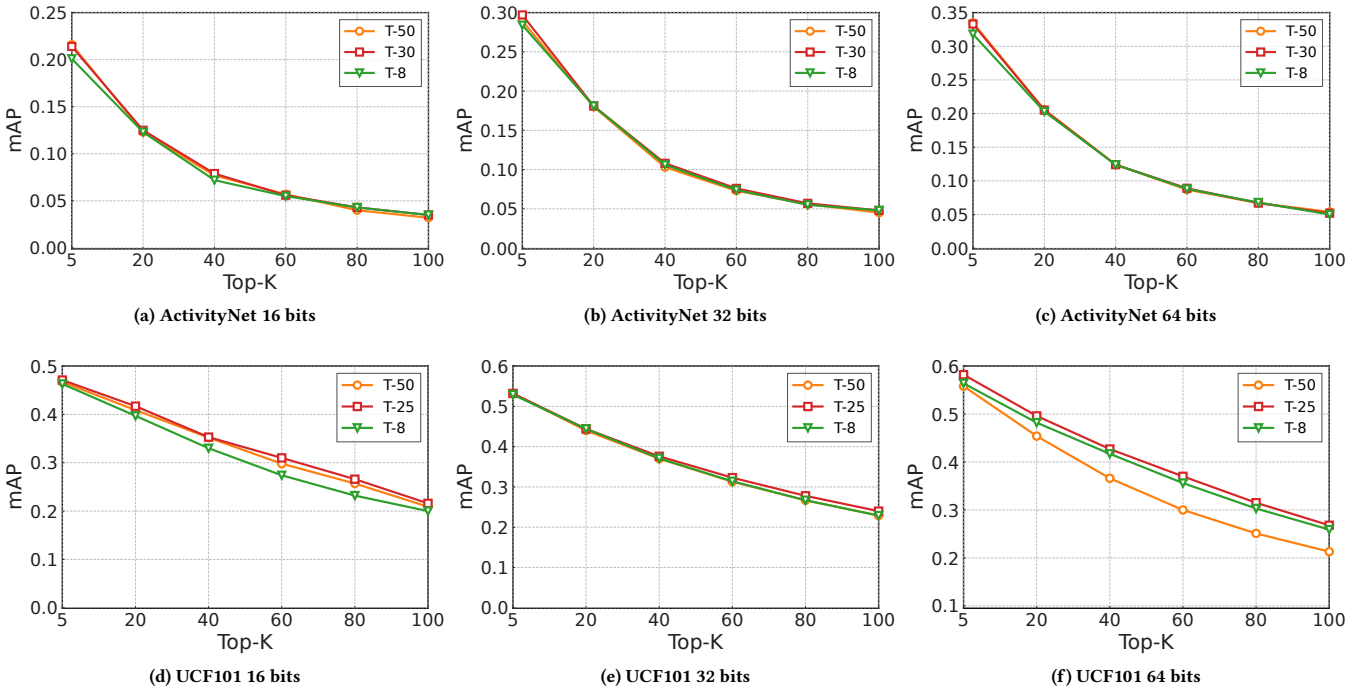
temporal encoder	K=5	K=20	K=40	K=60	K=80	K=100
CHAIN +LSTM	0.294	0.169	0.109	0.079	0.060	0.047
CHAIN +Transformer	0.333	0.205	0.124	0.089	0.067	0.052
CHAIN +MC-MLP	0.339	0.209	0.128	0.092	0.069	0.051



(a) ActivityNet 16 bits (b) ActivityNet 32 bits (c) ActivityNet 64 bits

(d) UCF101 16 bits (e) UCF101 32 bits (f) UCF101 64 bits

Figure 7: Comparison of retrieval performance using different 2D encoders in terms of mAP@K on two datasets.



(a) ActivityNet 16 bits (b) ActivityNet 32 bits (c) ActivityNet 64 bits

(d) UCF101 16 bits (e) UCF101 32 bits (f) UCF101 64 bits

Figure 8: Retrieval performance varying with different number of frames.