Learn from Unlabeled Videos for Near-duplicate Video Retrieval

Xiangteng He[†]
hexiangteng@pku.edu.cn
Wangxuan Institute of Computer Technology, Peking University
Beijing, China

Yulin Pan[†]
yanwen.pyl@alibaba-inc.com
Alibaba Group
Hangzhou, China

Mingqian Tang
mingqian.tmq@alibaba-inc.com
Alibaba Group
Hangzhou, China

Yiliang Lv
yiliang.lyl@alibaba-inc.com
Alibaba Group
Hangzhou, China

Yuxin Peng[*]
pengyuxin@pku.edu.cn
Wangxuan Institute of Computer Technology, Peking University
Beijing, China

ABSTRACT

Near-duplicate video retrieval (NDVR) aims to find the copies or transformations of the query video from a massive video database. It plays an important role in many video related applications, including copyright protection, tracing, filtering and etc. Video representation and similarity search are crucial to any video retrieval system. To derive effective video representation, most video retrieval systems require a large amount of manually annotated data for training, making it costly inefficient. In addition, most retrieval systems are based on frame-level features for video similarity searching, making it expensive both storage wise and search wise. To address the above issues, we propose a video representation learning (VRL) approach to effectively address the above shortcomings. It first effectively learns video representation from unlabeled videos via contrastive learning to avoid the expensive cost of manual annotation. Then, it exploits transformer structure to aggregate frame-level features into clip-level to reduce both storage space and search complexity. It can learn the complementary and discriminative information from the interactions among clip frames, as well as acquire the frame permutation and missing invariant ability to support more flexible retrieval manners. Comprehensive experiments on two challenging near-duplicate video retrieval datasets, namely FIVR-200K and SVD, verify the effectiveness of our proposed VRL approach, which achieves the best performance of video retrieval on accuracy and efficiency.

CCS CONCEPTS

• **Information systems → Video search**.

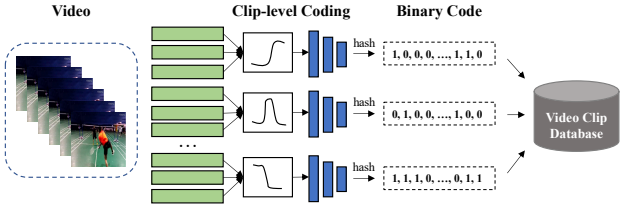

Figure 1: The illustration of video representation learning. Videos are split into shots via shot boundary detection, and further divided into clips at a fixed time interval. Then our VRL approach is applied to extract clip-level video representation. Finally, we binarize the clip-level features via hashing method for efficient retrieval.

KEYWORDS

Near-duplicate Video Retrieval, Video Representation Learning, Similarity Search

ACM Reference Format:
Xiangteng He, Yulin Pan, Mingqian Tang, Yiliang Lv, and Yuxin Peng. 2022. Learn from Unlabeled Videos for Near-duplicate Video Retrieval. In *Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '22), July 11–15, 2022, Madrid, Spain.* ACM, New York, NY, USA, 10 pages. https://doi.org/10.1145/3477495.3532010

1 INTRODUCTION

These days, we have witnessed dramatic increase in the volume of videos generated over internet. At the same time, we have also observed a large number of videos that essentially steal contents from others, making video copyright protection and filtering an important demand. Near-duplicate video retrieval (NDVR) aims to address the problem by identifying the copies or transformations of the query video from a large video database, which has drawn much attention [1–3]. It is one of the tasks in video retrieval, which also contain content-based video retrieval [4], video moment retrieval [5], video-text retrieval [6] and so on.

To design an efficient and effective near-duplicate video retrieval system, two components are important, i.e. video representation

†Equal contribution.
*Corresponding author.

SIGIR '22, July 11–15, 2022, Madrid, Spain.

ACM ISBN 978-1-4503-8732-3/22/07…$15.00
https://doi.org/10.1145/3477495.3532010

and similarity search. For video representation, most existing methods [7–9] apply supervised deep learning technologies to learn appropriate feature representation for accurate video content matching. It is costly and hard to learn the robust and powerful video representation in this way, due to: (1) A large amount of labeled videos are needed for the learning process. (2) The limited labeled videos restrict the ability of learning. So this is *the first issue* we should address in this paper.

For similarity search, most methods [8, 10–12] represent each video by a set of frame-level features, and the similarity between two videos are decided by the similarities between their frames followed by temporal alignment analysis, such as dynamic programming[11, 13], temporal network[12, 14, 15], and Temporal Hough Voting[16, 17]. The main shortcoming of these methods are of two folders. First, it needs to store all the frame-level features from all videos, making it *storage expensive*. Second, since the similarity measurement between two videos requires the similarity measurement between frames, making it *computationally expensive*. One common approach [7, 9] to address these limitations is to represent each video by a single vector (i.e. video-level features). Although these approaches help alleviate the problems of storage and computational cost, as pointed out in [18], they are insufficient to capture crucial details of individual videos, particularly for long videos. So this is *the second issue* we should address in this paper.

To address the above issues, we propose a video representation learning (VRL) approach. It leverages contrastive learning to learn video representation from large amounts of unlabeled videos, and exploits transformer structure to aggregate frame-level features into clip-level, as shown in Figure 1. More specifically, the key contributions of this work can be summarized as follow:

- **Frame-level encoding** is proposed to learn the frame-level representation with the pairs of the video frames and their transformations, which are automatically generated by temporal and spatial transformations, thus avoiding the high cost in manual annotation. It encodes the frame-level features via exploring the discriminative spatial structure with contrastive learning.Due to the self-generation of training data, our VRL approach can learn better frame-level representation from a large amount of unlabelled videos, leading to better generalization.
- **Clip-level encoding** is proposed to aggregate frame-level features into clip-level, leading to significant reduction in both storage space and search complexity. It can learns the complementary and variant information from the interactions among clip frames via self-attention mechanism, as well as acquires the frame permutation and missing invariant ability to handle the issue of missing frames, both of which increase the discrimination and robustness of the clip-level feature. Besides, it supports more flexible retrieval manners, such as clip-to-clip retrieval and frame-to-clip retrieval.

Comprehensive experiments on two challenging video retrieval datasets, namely FIVR-200K and SVD, verify the effectiveness of our VRL approach, which achieves the best performance of video retrieval on accuracy and efficiency. Compared with video-level methods, our VRL achieves the improvements of 30.6%, 28.2%, 21.3% mAPs on the DSVR, CSVR and ISVR tasks of FIVR-200K dataset, and 4.7% mAP on SVD dataset. Compared with frame-level methods, our VRL approach achieves comparable performance, and reduces about 78.7% of the feature storage cost and increases the retrieval speed by $\sim$ 25 times.

2 RELATED WORK

Existing video retrieval methods can be divided into two categories: frame-level retrieval methods and video-level retrieval methods.

2.1 Frame-level Retrieval Methods

These methods generally extract frame-level features using CNN, and retrieve related frames by approximate nearest neighbor search. Various post-processing methods[12, 15, 16, 19–21] have been proposed to aggregate the frame-to-frame similarity matrix to video similarity score. Jiang et al. propose Temporal Hough Voting [16] to find temporal alignments, which makes full use of the relative timestamp between matched frames. Tan et al. propose Graph-based Temporal Network [12] to detect the longest shared path between two compared videos. Hu and Lu [15] combine temporal network with a CNN+RNN feature encoder, to address the problem of partial copied detection. Another popular solution is based on Dynamic Programming(DP), which is applied to extract the biggest matched diagonal block from frame-to-frame similarity matrix, and tolerate limited horizontal and vertical movements for flexibility. Chou et al. [11] apply Bag-of-Words to represent frames, and propose m-pattern-based dynamic programming (mPDP) algorithm to localize near-duplicate segments and re-rank the retrieved videos. However, the above methods ignore exploiting spatial feature invariance, which is essential to video retrieval. Recently, Kordopatis-Zilos et al. [8] employ a region-level similarity calculation and aggregate region similarity matrix to frame similarities, which considers fine-grained spatial alignments and achieves high retrieval performance. These frame-level retrieval methods disregard the redundancy between successive frames, so that more computation cost will be needed, resulting in a low retrieval efficiency.

2.2 Video-level Retrieval Methods

Video-level retrieval methods encode the videos in video-level, and search for the k-nearest neighbors for the video-level feature of the query video in the embedding space. Various frame feature aggregation methods[7, 10, 22–25] have been used to obtain a single video-level representation. Liong et al. [26] propose temporal pooling layer to aggregate the successive frames by the means of average pooling. Kordopatis-Zilos et al. [10] extract individual frame features from intermediate CNN layers, and adopt Bag-of-Words to compress them into a video-level representation, so that video similarity can be measured by calculating the cosine distance between the two video-level representations. Furthermore, Kordopatis-Zilos et al. [7] aggregate frame features by the means of average pooling, and introduce Deep Metric Learning(DML) to learn an embedding by minimizing the distance between related videos and maximizing the distance between irrelevant ones. Hash codes based methods [23, 27–29] are also widely used to encode unified spatial-temporal representation from videos. Song et al. [29] capture the temporal relationship between frames using an encoder-decoder architecture. Li et al. [27] apply the binary codes to capture spatial-temporal

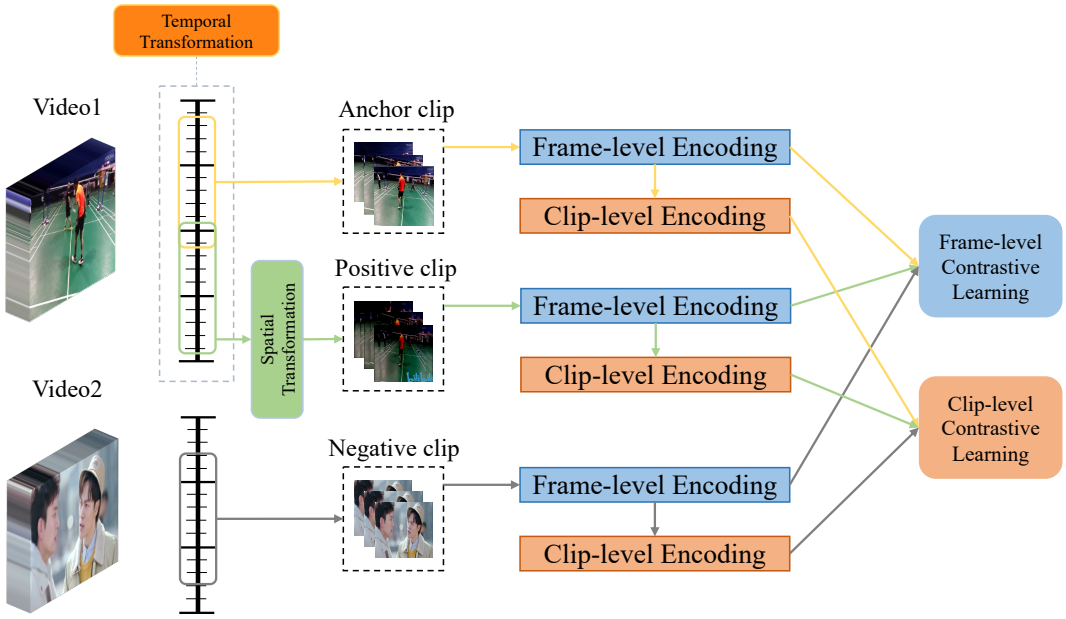

Figure 2: Overview of our VRL approach.

structure in a video by integrating the neighborhood attention mechanism into an RNN-based reconstruction scheme. However, these video-level retrieval methods generally perform worse than frame-level retrieval methods, which is mainly due to that single vector is hard to capture the entire spatio-temporal structure in a video sufficiently.

3 LEARN FROM UNLABELED VIDEOS

In this section, we present the proposed video representation learning (VRL) approach for efficient retrieval by reducing the expensive cost of manual annotation, storage space and similarity search. It mainly consists of two components: frame-level encoding and clip-level encoding, as shown in Figure 2. First, we automatically generate the frame or video clip pairs via temporal and spatial transformations. Then, we utilize these pairs as supervision to learn frame-level feature with contrastive learning. Finally, we aggregate the frame-level features into clip-level feature via self-attention mechanism, and increase the robustness via masked frame modeling.

3.1 Frame-level Encoding

Existing methods generally train their model with manual annotated video pairs. The more data, the better performance will be achieved [30]. However, the cost of annotation is too expensive to generate a large amount of training data. So the video representation learning is restricted to the limited volume of training data. Inspired by the advance of contrastive learning methods [31], we propose to learn the frame-level representation from large amounts of unlabeled videos with contrastive learning to break the restriction, and exploit the spatial invariant of representation to defense various video transformations.

3.1.1 Self-generation of Training Data. First, we automatically collect a large amounts of videos from the video website, and the details will be introduced in Sec. 4.2. Then, temporal and spatial transformations are sequentially performed on these clips to construct the training data.

(1) Temporal Transformation: As shown in the left part of Figure 3, given a video, we first uniformly sample N frames with a fixed time interval r to generate the anchor clip, denoted as $C = \{I^1, I^2, \cdots, I^N\}$. Then a frame I^m is randomly selected from the anchor clip as the identical content shared by anchor clip C and positive clip C_+. We regard the selected frame as the median frame of C_+, and uniformly sample $\frac{N-1}{2}$ frames forward and backward respectively, with a different sample time interval r_+. r is set to 1, and r_+ is set to from 0.5 to 2 randomly.

(2) Spatial Transformation: For each frame, we further perform spatial transformation. As shown in the right part of Figure 3, three types of spatial transformations are considered: *(a) Photometric transformation.* It includes the transformations of brightness, contrast, hue, saturation and gamma adjustment. *(b) Geometric transformation.* It includes the transformations of horizontal flip, rotation, crop, resize and translation. *(c) Editing transformation.* It includes the transformations of adding blurred background, adding logo, picture in picture and etc.. In training stage, we randomly select one transformation from each type of spatial transformation, and then apply them with randomly gray transformation and color constancy transformation on frames from positive clips in sequence to generate the new positive clips.

3.1.2 Spatial Structure Encoding. Since the supervised pairs are generated, we first utilize them to learn the video representation with frame-level contrastive loss. As shown in Figure 4, we adopts ResNet 50 [32] as the backbone, and then follow a convolutional

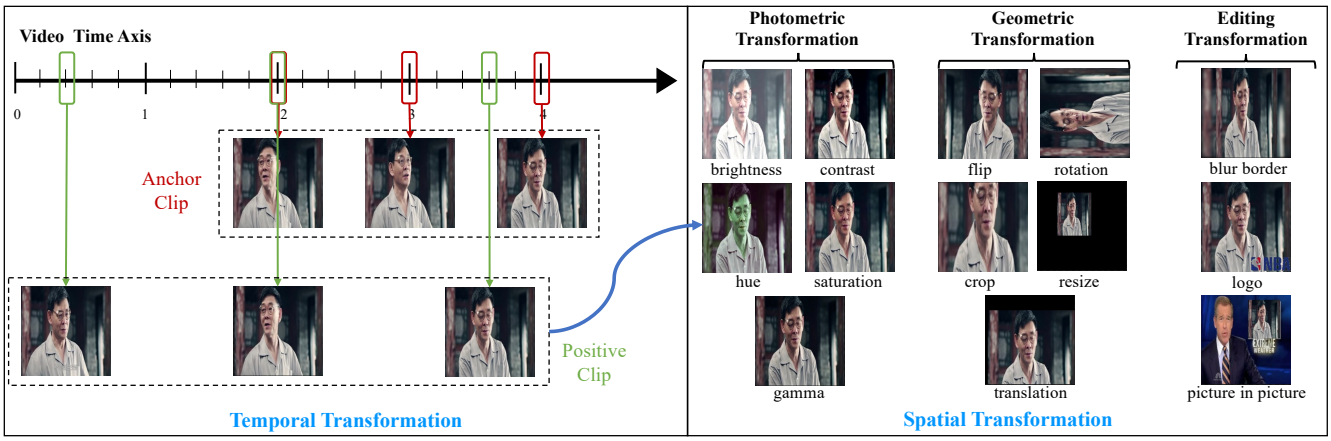

Figure 3: Illustration of self-generation of training data.

layer to reduce the channel number of the feature map, finally average pooling and L_2 normalization are applied to obtain the frame-level feature.

The goal of spatial structure encoding is to capture spatial discrimination from individual frames and ignore the impacts of various transformations, through minimizing the distance between features of the anchor clip frames and positive clip frames, as well as maximizing the distance between features of the anchor/positive clip frames and negative clip frames.

Specifically, given an anchor clip containing N frames $C = \{I^1, I^2, \cdots, I^N\}$, then a positive clip is generated via temporal and spatial transformations, denoted as $C_+ = \{I_+^1, I_+^2, \cdots, I_+^N\}$. We organize these frames in semantic-related pairs $\{(I^t, I_+^t)\}_{t=1}^N$. Then spatial structure encoding is employed to encode spatial discrimination from individual frames, which is formulated as

$$v = f_S(I) \tag{1}$$

Since a set of frame-level features $S_F = \{(v^t, v_+^t)\}_{t=1}^N$ is obtained, a contrastive learning is adopted to drive the features more discriminative and robust. The loss function is an adapted noise contrastive estimation loss [31], and its definition is as follow:

$$\begin{aligned} L_F = \frac{1}{N}\sum_{t=1}^{N} & -\mathbb{E}_{P_d}\log P(D=1|v^t, v_+^t) \\ & -(1-\mathbb{E}_{P_d})\log(1-P(D=1|v^t, v_+^t)) \end{aligned} \tag{2}$$

where P_d denotes the actual data distribution and $\mathbb{E}_{P_d} = 1$ indicates I^t and I_+^t share absolutely identical visual semantic. The probability of the encoded vectors v^t with v_+^t is from the data distribution $P(D=1|v^t, v_+^t)$ can be defined as :

$$P(D=1|v^t, v_+^t) = \frac{\exp({v^t}^{\mathrm{T}} v_+^t)}{\exp({v^t}^{\mathrm{T}} v_+^t) + \max\limits_{v_- \notin S_F} \exp({v^t}^{\mathrm{T}} v_-)} \tag{3}$$

where v_- indicates the feature of frame from the negative chip, which is semantic-irrelevant with anchor clip. It is noted that only the batch-hardest negative frame will contribute to the $P(D = 1|v^t, v_+^t)$, because the simple negative frames will decrease the discriminability of the learned feature.

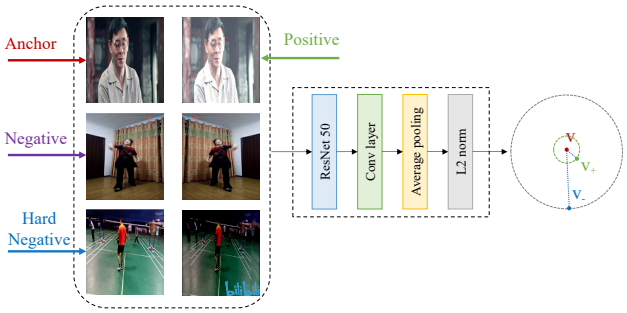

Figure 4: Model architecture.

3.2 Clip-level Encoding

Since the adjacent frames from one clip have the similar content, the frame-level features have high redundancy between each other, and the complementary information is not fully explored. Therefore, we aggregate the frame-level features into clip-level feature in this paper, to reduce both the storage space and search complexity. Specifically, given a clip, a set of frame-level features $\{v^1, v^2, \cdots, v^N\}$ are extracted through frame-level encoding, then aggregated into a single clip-level feature x, which is defined as follow:

$$x = f_C(v^1, v^2, \cdots, v^N) \tag{4}$$

To encode the clip-level feature, we propose an adapted Transformer [33], called clip-level set transformer network, whose architecture is shown in Figure 5. Instead of directly using Transformer to encode the clip-level feature, we apply the idea of set retrieval [34] in the clip-level encoding. It is noted that we only use one encoder layer with 8 attention heads, without position embedding. It enables our VRL approach has the abilities: (1) *More robust.* Increase the robustness of the learned clip-level features with the ability of

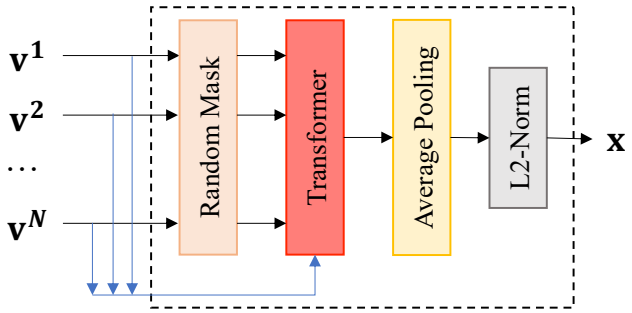

Figure 5: Architecture of our clip-level set transformer network.

Figure 6: Examples of videos collected from video website.

frame permutation and missing invariance. (2) *More flexible*. Support more retrieval manners, including clip-to-clip retrieval and frame-to-clip retrieval.

3.2.1 Temporal Structure Encoding. Similar with spatial structure encoding in Section 3.1.2, given a set of clip-level features $S_C = \{(x^b, x_+^b)\}_{b=1}^B$, where B is the number of clips in a batch, a clip-level constrastive learning is adopted. The loss function is defined as follow:

$$L_C(x, x_+) = \frac{1}{B}\sum_{b=1}^{B} -\mathbb{E}_{P_d}\log P(D=1|x^b, x_+^b) - (1-\mathbb{E}_{P_d})\log(1-P(D=1|x^b, x_+^b)) \tag{5}$$

where P_d denotes the actual data distribution and $\mathbb{E}_{P_d}$ is set to 1 indicates the anchor clip and positive clip share absolutely identical visual semantic. $P(D=1|x^b, x^b{}_+)$ denotes the posterior probability that x with x_+ is from the actual data distribution, its definition is similar with Equation (3). Clip-level encoding can learn the complementary information from the frames of the video clip via self-attention mechanism of Transformer, and hence the discimination of features via attentively seeing the frames.

3.2.2 Masked Frame Modeling. To increase the robustness of the learned clip-level features, we treat the frames of one clip as a set, and randomly mask some frames in clip-level encoding. For a given clip C, we randomly drop some frames to generate a new clip C'. Its goal is to eliminate the influence of frame blur or clip cut, and drive the model to have the ability that use any combination of any frames in the clip can retrieval its corresponding clips.

Specially, given a clip-level feature x, its new feature after conducting masked frame modeling is denoted as x'. Similarly, the corresponding positive clip-level feature and its new feature are denoted as x_+ and x'_+. Therefore, we need to learn from the following loss functions: $L_C(x, x'_+)$ and $L_C(x', x_+)$. So the final loss function of clip-level set transformer network is defined as follows:

$$L_C = L_C(x, x_+) + L_C(x, x'_+) + L_C(x', x_+) \tag{6}$$

3.3 Video Similarity Calculation

Instead of directly using the whole video frames or divide the video into clips with fixed time interval, we first conduct shot boundary detection for each video to segment the videos into shots, and then divide the shots into clips at a fixed time interval, i.e N seconds. In this way, we can guarantee the frames within the same clip contain similar contents, so that clip-level encoding can reduce the redundancy rather than losing import information. Second, the sequence of successive frames is passed through the clip-level set transformer network to generate the clip-level feature. Finally, the clip-level feature is binarized by IsoHash[35] to further reduce the storage cost and search cost. When retrieving, we measure the clip-to-clip similarities with hamming distance. Given an $M \times N$ clip-to-clip similarity matrix, the video similarity score can be calculated as follows:

$$Sim = \frac{1}{M}\sum_{i=1}^{M}\max_{j\in[1,N]} CS(i,j) \tag{7}$$

where $CS(i, j)$ denotes the similarity score between clip i and j, and it is calculated as follows:

$$CS(i,j) = \max_{k\in K}\mathcal{H}(i,k) - \mathcal{H}(i,j) \tag{8}$$

in which K indicates the entire clip set and $\mathcal{H}(\cdot,\cdot)$ indicates the hamming distance calculation.

4 EXPERIMENTS

4.1 Implementation Details

At training phase, we adopt default SGD optimizer with the batch size of 64. We use a weight decay of 5×10^{-6} with a momentum of 0.9 and set the initial learning rate as 0.0001. The model is trained for 5 epochs, and the learning rate learning rate is divided by 10 after the first epoch.

4.2 Datasets

Our VRL approach is trained on our constructed Self-Transformation dataset, and performs evaluations on two challenging near-duplicate video retrieval datasets, namely FIVR-200K and SVD. The detailed information is introduced as follows:

- **Self-Transformation** is constructed by randomly collecting videos from video website [1], as shown in Figure 6. It

[1] https://www.youku.com/

Feature	Methods	Feature Dim/#bits	DSVR	CSVR	ISVR
Video-level	HC[36]	-	0.265	0.247	0.193
	DML[7]	500D	0.398	0.378	0.309
	TCA_c[9]	2048D	0.570	0.553	0.473
Frame-level	CNN-L[10]	4096D	0.710	0.675	0.572
	PPT[11]	4096D	0.775	0.740	0.632
	TN[12]	-	0.724	0.699	0.589
	TCA_f[9]	2048D	0.877	0.830	0.703
	VisiL[8]	9x3840D	0.892	0.841	0.702
	VRL_f	**512 bits**	**0.900**	**0.858**	**0.709**
Clip-level	**VRL**	**512 bits**	**0.876**	**0.835**	**0.686**

Table 1: Comparisons with state-of-the-art methods on all three tasks of FIVR-200K dataset.

consists of 3,000 hours' videos, and temporal and spatial transformations are performed at training stage.

- **FIVR-200K** [2] consists of 225,960 videos and 100 queries. It is constructed for fine-grained incident video retrieval, including three retrieval tasks: (1) Duplicate scene video retrieval (DSVR) is to retrieval the videos sharing at least one scene that captured by the same camera, regardless of any transformation. (2) Complementary scene video retrieval (CSVR) is to retrieval the videos containing part of the same spatio-temporal segment with different views. (3) Incident scene video retrieval (ISVR) is to retrieval the videos capturing the same event without the same overlapped saptio-temporal segment. We evaluate our VRL approach on all the three tasks to verify its effectiveness.
- **SVD** [1] is constructed for short video retrieval task. It consists of 562,013 short videos with the duration less than 60 seconds. It contains 1,206 query videos, and over 30,000 labelled videos in which the negatives have extremely similar but different appearance. Besides, there are more than 500,000 hard negative unlabelled distraction videos to increase the retrieval difficulty.

4.3 Evaluation Metric

Following [1, 2], we apply the mean average precision (mAP) score to evaluate the video retrieval performance. We first calculate average precision (AP) score for each query, and then calculate their mean value as mAP score.

4.4 Comparisons with State-of-the-art Methods

In this subsection, experimental results and analyses of comparing our proposed VRL approach with the state-of-the-art methods on FIVR-200K and SVD datasets are presented, which are shown in Table 1 and Table 2. It is noted that we evaluate our VRL approach on all the three tasks of FIVR-200K dataset, including DSVR, CSVR, ISVR.

4.4.1 Comparisons with Frame-level Retrieval Methods. We then compare our VRL approach with 6 frame-level retrieval methods, which are briefly introduced as follows:

- CNN-L and CNN-V [10] are proposed to convert multiple intermediate CNN features into one vector via layer and vector aggregation schemes respectively.

Feature	Methods	Feature Dim/#bits	Top-100 mAP
Video-level	DML[7]	500D	0.813
Frame-level	CNN-L[10]	4096D	0.610
	CNN-V[10]	4096D	0.251
	VRL_f	**512 bits**	**0.871**
Clip-level	**VRL**	**512 bits**	**0.860**

Table 2: Comparisons with state-of-the-art methods on SVD dataset.

- PPT [9] is a spatio-temporal pattern-based method under the hierarchical filter-and-refine framework.
- Temporal Network (TN) [12] is proposed to detect the longest shared path between two videos.
- Temporal Context Aggregation (TCA) [9] is proposed to learn a single video vector by aggregating frame-level features with self-attention. In comparisons with frame-level retrieval methods, we report its result with frame-level feature.
- VisiL [8] is proposed to calculate video-to-video similarity from refined frame-to-frame similarity matrices.

Compare with frame-level retrieval approach, our VRL approach outperforms all state-of-the-art methods except VisiL. It is noted that VisiL adopts a region-aligned matching scheme, which is impractical for large-scale retrieval task due to its low efficiency. While our VRL approach still achieves comparable retrieval performance with VisiL under the situation that only using binary codes and no any re-ranking process. Furthermore, when encoding the videos in frame-level features, our VRL_f approach can achieve better retrieval performance than VisiL without any complex calculation. It is mainly because our VRL approach can boost the discrimination of the features due to the learning from the self-generated training data, which has strong power in representation learning. Importantly, no annotated video pairs are needed, which efficiently reduces the expensive cost of manual annotation.

4.4.2 Comparisons with Video-level Retrieval Methods. We first compare our VRL approach with 3 video-level retrieval methods, which are briefly introduced as follows:

- Hashing Codes (HC) [36] is proposed to learn a group of hash functions based on frame-level features, and then combine the hash codes into a single video vector.
- Deep Metric Learning (DML) [7] is proposed to early or late fuse the frame-level features into a single video vector, which is then fine-tuned by deep metric learning.
- Temporal Context Aggregation (TCA) [9], whose result with video-level feature is reported.

Compare with video-level retrieval methods, clip-level retrieval methods needs more storage and search cost. To reduce these costs, we utilize hash codes and measure hamming distances while other methods use floats and measure Euclidean or Cosine distances. Even so, our VRL approach achieves significant improvements by 30.6%, 28.2%, 21.3% mAPs on the DSVR, CSVR and ISVR tasks of FIVR-200K dataset, as well as 4.7% mAP on SVD dataset, which are shown in Table 1 and Table 2. It are mainly because: (1) Clip-level feature encoding can extract more abundant and complementary information from the interactions among clip frames. (2) Clip-level set transformer network can aggregate the frame features in one clip considering their different roles, which takes full advantage of each frame's discrimination, and eliminates the redundancy between the adjacent frames. Besides, it acquires the frame permutation and missing invariant ability with masked frame modeling.

Feature	Storage Space	Search Complexity
Frame-level	1720.32 MB	$O(M \times N)$
Clip-level	**366.98 MB**	$\sim \frac{1}{25}\boldsymbol{O(M \times N)}$

Table 3: Reduction of storage and search cost on SVD dataset.

4.5 Effectiveness of Reducing Storage and Search Cost

To verify the effectiveness of our proposed VRL approach on reducing the storage and search cost, we compare the storage spaces and search complexities between frame-level retrieval and clip-level retrieval on SVD dataset. As shown in Table 3, the storage of the frame-level features cost 1720.32 MB, while clip-level features only cost 366.98 MB, reducing the storage cost by 78.7%. It is mainly because that our VRL approach first segments the videos into shots, and then divides the shots into clips, finally encodes the clips to represent the videos.

Suppose that the SVD dataset has m queries and n videos, and they are encoded by M frame-level features and N frame-level features respectively. So it needs $O(M \times N)$ similarity computation. However, depend on the above analyses, they can be encoded by $\sim \frac{5}{M}$ and $\sim \frac{5}{N}$ clip-level features respectively, so only $O(\sim \frac{M}{5} \times \sim \frac{N}{5}) = \sim \frac{1}{25}O(M \times N)$ similarity computation is needed. In other words, our VRL approach increases the retrieval speed by ~ 25 times, which verifies that clip-level video retrieval is an efficient retrieval paradigm to reduce the storage cost and search cost.

4.6 Exploration of Flexible Retrieval Manners

As mentioned above, clip-level set transformer network provides more flexible retrieval manners, i.e. clip-to-clip retrieval and frame-to-clip retrieval. So we explore their retrieval performance on SVD dataset, as shown in Table 4. The videos in database are all encoded in clip-level features, only different in query encoding. We can see that use more fine-grained features (i.e. frame-level) can achieve better retrieval performance, which further verifies the effectiveness of clip-level encoding with masked frame modeling, which can driven our VRL approach learn both clip-level and frame-level features. With more flexible retrieval manners, our VRL approach has more application prospects.

Query	Database	Top-100 mAP
Clip-level	Clip-level	0.860
Frame-level	Clip-level	0.871

Table 4: Results of different retrieval manners.

Methods	Transformations			DSVR	CSVR	ISVR
	PT	GT	ET			
VRL$_f$	✓	✓	✓	**0.900**	**0.858**	**0.709**
A	✓	✓		0.868	0.818	0.673
B	✓		✓	0.881	0.825	0.662
C		✓	✓	0.868	0.815	0.649

Table 5: Impacts of different spatial transformations on FIVR-200K dataset.

4.7 Ablation Study

Detailed experiments are performed to further verify the effectiveness of our VRL approach in the following aspects:

4.7.1 Effectiveness of Self-generation of Training Data. We directly utilize the frame-level features to perform video retrieval, results are shown in Table 1 and Table 2 as "VRL$_f$". It outperforms than state-of-the-art methods on both two datasets, which verifies the effectiveness of the spatial structure encoding from the large amounts of videos. Due to self-generation of training data, we can generate the training data as much as we want, which breaks the restriction of the expensive manual annotation cost.

Besides, we further evaluate the impact of each transformation on the retrieval performance of frame-level encoding. The results of three tasks on FIVR-200K dataset are shown in Table 5, where "PT", "GT" and "ET" denote photometric transformation, geometirc transformation and editing transformation respectively, as well as the experiments of "A", "B" and "C" denote training without editing transformation, geometric transformation and photometric transformation respectively. We can observe that "VRL$_f$" with all the three types of transformations achieves the best performance, which verifies that each transformation plays an irreplaceable role on video representation learning. They provide rich supervision information to drive the model to approximate the real data distribution, which make the learned representation spatial-temporal invariant.

In addition, we also show the examples of retrieved results in Figure 7, where (a) is from SVD dataset, and (b) and (c) are from FIVR-200K dataset. We can observe that even the copied videos are

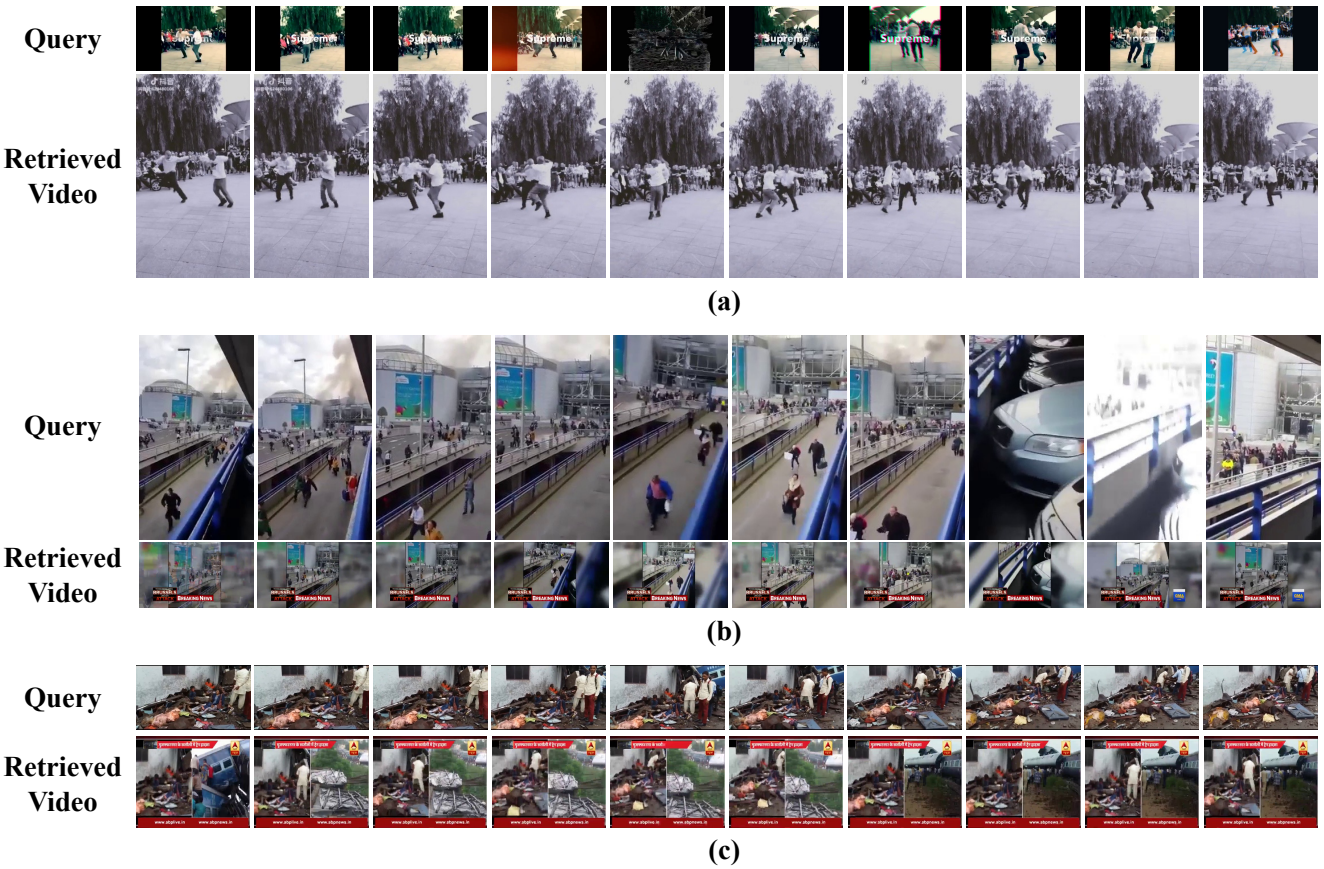

Figure 7: Examples of retrieved results.

very different from the original video with complex transformations, our VRL approach can also find them from the large scale of database, which verifies its effectiveness.

4.7.2 Effectiveness of Clip-level Set Transformer Network. In the clip-level encoding, we propose the clip-level set transformer network to encode the clip-level features. To verify its effectiveness, we compare it with the NetVLAD [37], which aggregates the frame-level features via a new generalized VLAD layer. The results are shown in Table 6. We can observe that our VRL approach achieves better performance than NetVLAD method. It is because that the complementary and variant information can be learned from the interactions among clip frames via self-attention mechanism, and the masked frame modeling can make our VRL approach has the ability of frame permutation and missing invariance.

4.7.3 Effectiveness of Masked Frame Modeling. We evaluate the effectiveness of clip-level encoding with masked frame modeling on FIVR-200K and SVD datasets. Results are shown in Table 7, where "CE" and "MFM" denote clip-level encoding and masked frame modeling respectively. Clip-level encoding with masked frame modeling coerces the Transformer to learn the correlations between the anchor clip with missing information and positive clip, as well as the anchor clip and positive clip with missing information, which makes the transformer more robust and not sensitive to the frame missing. So it improves the discrimination and robustness of the learned clip-level feature, and achieves better performance. Besides, due to masked frame modeling, our VRL approach can support more flexible retrieval manners, i.e. clip-to-clip retrieval and frame-to-clip retrieval. The results can be found in Table 4.

Methods	SVD	FIVR-200K		
		DSVR	CSVR	ISVR
NetVLAD [37]	0.706	0.690	0.636	0.503
VRL	**0.860**	**0.876**	**0.835**	**0.686**

Table 6: Comparison between our clip-level encoding and NetVLAD.

Methods	SVD	FIVR-200K		
		DSVR	CSVR	ISVR
CE	0.854	0.870	0.834	**0.687**
CE w/ MFM	**0.860**	**0.876**	**0.835**	0.686

Table 7: Effectiveness of clip-level encoding with masked frame modeling.

Clip Length	SVD	FIVR-200K		
		DSVR	CSVR	ISVR
4s	0.861	0.883	0.841	0.693
6s	0.867	0.881	0.837	0.688
8s	0.860	0.876	0.835	0.686

Table 8: Impact of clip length on clip-level encoding.

4.7.4 Impact of Clip Length. Besides, we evaluate the impact of clip length to the retrieval performance of clip-level set transformer network. Table 8 shows the results of different clip length settings on FIVR-200K and SVD datasets. We can observe that our clip-level set transformer network is not very sensitive to the clip lengths. It is mainly because that we apply masked frame modeling in clip-level encoding, which drives the model to have the ability that any combination of any frames in the clip can retrieval its corresponding clips. So to balance the retrieval accuracy and efficiency, we set the clip length as 8s in our experiments.

5 CONCLUSION

This paper proposes the VRL approach to encode the video in clip-level representation with contrastive learning to reduce the expensive cost of manual annotation, storage space and similarity search . It consists of two components: (1) Frame-level encoding is to learn the discrimination and robustness of the learned feature with self-generation of training data, which automatically generate the pairs of the videos and their transformations as supervision information, reducing the heavy labor consumption in annotating. (2) Clip-level encoding is to reduce the redundancy of the frames in a clip, as well as learn the complementary and discriminative information from the interactions among clip frames. Besides, clip-level encoding with masked frame modeling make the model frame permutation and missing invariant, and support more flexible retrieval manners. Comprehensive experiments on two challenging video retrieval datasets, namely SVD and FIVR-200K, verify the effectiveness of our VRL approach, which achieves the best performance of video retrieval on accuracy and efficiency.

The future works will lie in two aspects: (1) How to design more efficient self-generation strategy? (2) How to transfer the motion information to RGB frames in training, but only use RGB frames in retrieval. Both of them will be explored to further improve the video retrieval performance.

6 ACKNOWLEDGMENTS

This work was supported by the grants from the National Natural Science Foundation of China (61925201, 62132001, U21B2025), the National Key R & D Program of China (2021YFF0901502), and by Alibaba Group through Alibaba Research Program.

REFERENCES

[1] Qing-Yuan Jiang, Yi He, Gen Li, Jian Lin, Lei Li, and Wu-Jun Li. Svd: A large-scale short video dataset for near-duplicate video retrieval. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, pages 5281–5289, 2019.

[2] Giorgos Kordopatis-Zilos, Symeon Papadopoulos, Ioannis Patras, and Ioannis Kompatsiaris. Fivr: Fine-grained incident video retrieval. *IEEE Transactions on Multimedia (TMM)*, 21(10):2638–2652, 2019.

[3] Zhen Han, Xiangteng He, Mingqian Tang, and Yiliang Lv. Video similarity and alignment learning on partial video copy detection. In *Proceedings of the 29th ACM International Conference on Multimedia (ACM MM)*, pages 4165–4173, 2021.

[4] Xiangming Mu. Content-based video retrieval: Does video's semantic visual feature matter? In *Proceedings of the 29th annual international ACM SIGIR conference on Research and Development in Information Retrieval (ACM SIGIR)*, pages 679–680, 2006.

[5] Feng He, Qi Wang, Zhifan Feng, Wenbin Jiang, Yajuan Lü, Yong Zhu, and Xiao Tan. Improving video retrieval by adaptive margin. In *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval (ACM SIGIR)*, pages 1359–1368, 2021.

[6] Peng Wu, Xiangteng He, Mingqian Tang, Yiliang Lv, and Jing Liu. Hanet: Hierarchical alignment networks for video-text retrieval. In *Proceedings of the 29th ACM International Conference on Multimedia (ACM MM)*, pages 3518–3527, 2021.

[7] Giorgos Kordopatis-Zilos, Symeon Papadopoulos, Ioannis Patras, and Yiannis Kompatsiaris. Near-duplicate video retrieval with deep metric learning. In *Proceedings of the IEEE International Conference on Computer Vision Workshops (ICCVW)*, pages 347–356, 2017.

[8] Giorgos Kordopatis-Zilos, Symeon Papadopoulos, Ioannis Patras, and Ioannis Kompatsiaris. Visil: Fine-grained spatio-temporal video similarity learning. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, pages 6351–6360, 2019.

[9] Jie Shao, Xin Wen, Bingchen Zhao, and Xiangyang Xue. Temporal context aggregation for video retrieval with contrastive learning. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 3268–3278, 2021.

[10] Giorgos Kordopatis-Zilos, Symeon Papadopoulos, Ioannis Patras, and Yiannis Kompatsiaris. Near-duplicate video retrieval by aggregating intermediate cnn layers. In *International Conference on Multimedia Modeling (MMM)*, pages 251–263. Springer, 2017.

[11] Chien-Li Chou, Hua-Tsung Chen, and Suh-Yin Lee. Pattern-based near-duplicate video retrieval and localization on web-scale videos. *IEEE Transactions on Multimedia (TMM)*, 17(3):382–395, 2015.

[12] Hung-Khoon Tan, Chong-Wah Ngo, Richard Hong, and Tat-Seng Chua. Scalable detection of partial near-duplicate videos by visual-temporal consistency. In *Proceedings of the 17th ACM International Conference on Multimedia (ACM MM)*, pages 145–154, 2009.

[13] Hao Liu, Qingjie Zhao, Hao Wang, Peng Lv, and Yanming Chen. An image-based near-duplicate video retrieval and localization using improved edit distance. *Multimedia Tools and Applications (MTA)*, 76(22):24435–24456, 2017.

[14] Yu-Gang Jiang and Jiajun Wang. Partial copy detection in videos: A benchmark and an evaluation of popular methods. *IEEE Transactions on Big Data (TBD)*, 2(1):32–42, 2016.

[15] Yaocong Hu and Xiaobo Lu. Learning spatial-temporal features for video copy detection by the combination of cnn and rnn. *Journal of Visual Communication and Image Representation (JVCIR)*, 55:21–29, 2018.

[16] Yu-Gang Jiang, Yudong Jiang, and Jiajun Wang. Vcdb: a large-scale database for partial copy detection in videos. In *European Conference on Computer Vision (ECCV)*, pages 357–371. Springer, 2014.

[17] Matthijs Douze, Hervé Jégou, and Cordelia Schmid. An image-based approach to video copy detection with spatio-temporal post-filtering. *IEEE Transactions on Multimedia (TMM)*, 12(4):257–266, 2010.

[18] Jingkuan Song, Yi Yang, Zi Huang, Heng Tao Shen, and Richang Hong. Multiple feature hashing for real-time large scale near-duplicate video retrieval. In *Proceedings of the 19th ACM International Conference on Multimedia (ACM MM)*, pages 423–432, 2011.

[19] Yang Feng, Lin Ma, Wei Liu, Tong Zhang, and Jiebo Luo. Video re-localization. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 51–66, 2018.

[20] Lorenzo Baraldi, Matthijs Douze, Rita Cucchiara, and Hervé Jégou. Lamv: Learning to align and match videos with kernelized temporal layers. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 7804–7813, 2018.

[21] Jérôme Revaud, Matthijs Douze, Cordelia Schmid, and Hervé Jégou. Event retrieval in large video collections with circulant temporal encoding. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2459–2466, 2013.

[22] Kaiyang Liao, Hao Lei, Yuanlin Zheng, Guangfeng Lin, Congjun Cao, Mingzhu Zhang, and Jie Ding. Ir feature embedded bof indexing method for near-duplicate video retrieval. *IEEE Transactions on Circuits and Systems for Video Technology (TCSVT)*, 29(12):3743–3753, 2018.

[23] Yang Cai, Linjun Yang, Wei Ping, Fei Wang, Tao Mei, Xian-Sheng Hua, and Shipeng Li. Million-scale near-duplicate video retrieval system. In *Proceedings of the 19th ACM International Conference on Multimedia (ACM MM)*, pages 837–838, 2011.

[24] Zhanning Gao, Gang Hua, Dongqing Zhang, Nebojsa Jojic, Le Wang, Jianru Xue, and Nanning Zheng. Er3: A unified framework for event retrieval, recognition and recounting. In *Proceedings of the IEEE Conference on Computer Vision and*

Pattern Recognition (CVPR), pages 2253–2262, 2017.

[25] Xiao Wu, Alexander G Hauptmann, and Chong-Wah Ngo. Practical elimination of near-duplicates from web video search. In *Proceedings of the 15th ACM International Conference on Multimedia (ACM MM)*, pages 218–227, 2007.

[26] Venice Erin Liong, Jiwen Lu, Yap-Peng Tan, and Jie Zhou. Deep video hashing. *IEEE Transactions on Multimedia (TMM)*, 19(6):1209–1219, 2016.

[27] Shuyan Li, Zhixiang Chen, Jiwen Lu, Xiu Li, and Jie Zhou. Neighborhood preserving hashing for scalable video retrieval. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, pages 8212–8221, 2019.

[28] Yanbin Hao, Tingting Mu, Richang Hong, Meng Wang, Ning An, and John Y Goulermas. Stochastic multiview hashing for large-scale near-duplicate video retrieval. *IEEE Transactions on Multimedia (TMM)*, 19(1):1–14, 2016.

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

[30] Chen Sun, Abhinav Shrivastava, Saurabh Singh, and Abhinav Gupta. Revisiting unreasonable effectiveness of data in deep learning era. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, pages 843–852, 2017.

[31] Aaron van den Oord, Yazhe Li, and Oriol Vinyals. Representation learning with contrastive predictive coding. *arXiv preprint arXiv:1807.03748*, 2018.

[32] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 770–778, 2016.

[33] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in Neural Information Processing Systems (NeurIPS)*, pages 5998–6008, 2017.

[34] Yujie Zhong, Relja Arandjelovic, and Andrew Zisserman. Compact deep aggregation for set retrieval. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 0–0, 2018.

[35] Weihao Kong and Wu-Jun Li. Isotropic hashing. In *Advances in Neural Information Processing Systems (NeurIPS)*, pages 1646–1654, 2012.

[36] Jingkuan Song, Yi Yang, Zi Huang, Heng Tao Shen, and Jiebo Luo. Effective multiple feature hashing for large-scale near-duplicate video retrieval. *IEEE Transactions on Multimedia (TMM)*, 15(8):1997–2008, 2013.

[37] Relja Arandjelovic, Petr Gronat, Akihiko Torii, Tomas Pajdla, and Josef Sivic. Netvlad: Cnn architecture for weakly supervised place recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5297–5307, 2016.