3D-CSL: SELF-SUPERVISED 3D CONTEXT SIMILARITY LEARNING FOR NEAR-DUPLICATE VIDEO RETRIEVAL

Rui Deng*, Qian Wu*, Yuke Li†

NetEase Yidun AI Lab, Hangzhou, China
{dengrui01, wuqian05, liyuke}@corp.netease.com

ABSTRACT

In this paper, we introduce 3D-CSL, a compact pipeline for Near-Duplicate Video Retrieval (NDVR), and explore a novel self-supervised learning strategy for video similarity learning. Most previous NDVR methods depend a lot on pair-wise labeled data, so that be limited by the scale of datasets and cannot optimize complex but efficient backbones, e.g., 3D transformers. In order to break this limitation, we explore the self-supervised similarity learning for the NDVR task and propose FCS loss, a novel triplet loss, and ShotMix, a novel video-specific augmentation, which enhances the self-supervised video similarity learning significantly. With this premise, the compact 3D pipeline we propose shows a great advantage in extracting global spatiotemporal dependencies in videos and achieves the best balance between efficiency and effectiveness. Furthermore, we also propose PredMAE to pretrain the 3D transformer with video prediction task as a pretext task to boost the downstream NDVR task without any human labels. The experiments on FIVR-200K and CC_WEB_VIDEO demonstrate the superiority and reliability of our method, which achieves the state-of-the-art performance on clip-level NDVR. Code is released in https://github.com/dun-research/3D-CSL.

Index Terms— Self-supervised Learning, Near-Duplicate Video Retrieval, Transformer

1. INTRODUCION

With the increasing proportion of video data on the Internet, video retrieval algorithms play a critical role in various scenarios, such as recommendation, search, de-duplication, etc. Previous methods pre-extract spatial features frame-by-frame with 2D networks and can be divided into frame-level/fine-grained and video-level/coarse-grained video retrieval. Frame-level methods [1–4] finely compare every pair of frames and compute similarities between two videos by designed mechanisms. The retrieval processes of frame-level methods are usually slow due to the fine-grained mechanism and cost a large amount of storage space because all the frame features must be stored. Video-level methods are proposed to further aggregate frame-level features so that every video can be represented by only one vector [5,6], hash code [7–9], or other forms of indexes [10,11], which are compact but may be insufficient to describe the details of the video. There is a trade-off between efficiency and effectiveness. So VRL [12] proposes to use clip-level features to represent a fixed-length video segment. Besides, temporal context information is essential for reducing the redundant information between frames and acquiring a determinate representation. Although some of the above methods [2,12] design modules to extract the temporal correlations between frame features, spatiotemporal dependencies are partially lost while extracting the frame features separately.

Therefore, this paper proposes a compact 3D clip-level pipeline to address the abovementioned problems. At first, we adopt the clip-level retrieval scheme as VRL [12] and split a video into several clips. And then, we extract spatiotemporal video features by a 3D transformer instead of frame-level spatial features. The final similarities are computed with these clip features, with no additional modules needed for obtaining temporal context, ensuring an extremely compact pipeline for video retrieval.

For training the video retrieval networks, most of the previous works [1,2,5] use pair-wise methods to supervise model optimization. Some [1,5] use triplet loss, which covers limited correlations and highly depends on sampling hard negative pairs. Also, pair-wise video annotations are very expensive, so their training datasets are usually small (e.g. the annotated core set of VCDB [14]). Thus TCA [2] takes advantage of more negative samples by constructing a memory bank referring to contrastive learning [15,16]. VRL [12] further proposes using large-scale unlabeled data through self-supervised learning, generating deformations of the anchor as positive samples with various spatial and temporal transforms to get rid of the dependence on annotations. They all use InfoNCE loss [17], a common loss in contrastive learning. However, InfoNCE loss penalizes all sample pairs equally. In contrast, hard positives (positive sample pairs with small similarities) and hard negatives (negative sample pairs with large similarities) samples should be optimized much more. So Circle loss [18] re-weights each similarity to highlight the hard samples. Multi-similarity loss (MS loss) [19] takes account of three types of similarities and sets a principle to mine the most informative sample pairs for optimization. However, the rel-
2. THE COMPACT VIDEO RETRIEVAL PIPELINE

A Compact 3D pipeline. As shown in Fig.2, we adopt a 3D backbone (TimeSformer [24] is used in this paper) as the spatiotemporal feature extractor to avoid losing temporal information, which happens when extracting frame features separately. In order to balance the retrieval efficiency and effectiveness, we extract clip-level features from video segments, which matches the scheme of the 3D video feature extractor well. Formally, the input video \( V \) is split into \( N \) clips and \( F \) frames are uniformly sampled from each clip. The 3D extractor takes clips \( X \in \mathbb{R}^{N \times F \times H \times W \times D} \) as input and learns the spatial, temporal and spatiotemporal correlations among frames all at once. The output \( Y \in \mathbb{R}^{N \times D} \) is the representations with Dimension \( D \) for the \( N \) input clips.

TopK Chamfer Similarity. The similarity between two videos should be measured considering all pairs of clips, and Chamfer Similarity (CS) is generally adopted. We denote the representations of two clip sequences as \( A = [a_1, a_2, ..., a_n] \in \mathbb{R}^{n \times D} \) and \( B = [b_1, b_2, ..., b_m] \in \mathbb{R}^{m \times D} \). The \( \text{Sim}_{CS} \) is computed as the mean of all the \( n \times m \) similarities \( \{\max_{j \in \{1, m\}}(\text{Sim}_{ai, bj})\}_{i=1}^{n} \). However, some of the maximum values are actually not large enough because the pairs of clips \( \{(a_i, b_j)\}_{i=1}^{n} \) are all dissimilar, which may lead to a smaller CS similarity than we expect. Thus, we generally focus on the most similar pair of clips [25], so we develop the CS to TopK-CS, where only the top \( k \) max similarities of \( A \) are averaged to get the final \( \text{Sim}_{topKCS} \) (we set \( k = 3 \) in this paper):

\[
\text{Sim}_{topKCS} = \frac{1}{k} \sum_{i=1}^{k} \text{sort}_{top}(\max_{j}(a_i b_j^T))
\]

(1)

The 3D pipeline discards the complex temporal context learning modules in previous works [1, 2, 12], and the TopK-CS can be computed fast with only one \( D \)-dimension matrix multiplication, which makes the retrieval process fast.

3. SELF-SUPERVISED 3D CONTEXT SIMILARITY LEARNING STRATEGY

Supervised similarity learning methods for NDVR task generally need positive/negative-pair annotations, which are very expensive. Thus, we propose a Self-Supervised 3D Context Similarity Learning method and enhance it with a novel triplet loss. Flipped Clip Similarity Loss (FCS loss) and a novel video-specific augmentation, ShotMix, as described in Sec.3.1. Besides, the 3D pipeline is more difficulty to be optimized than 2D pipelines due to more parameters. We further propose a self-supervised pretrained strategy, Predictive Masked Autoencoder (PredMAE), to reduce the optimization difficulty of the similarity learning in Sec.3.2.

3.1. Self-supervised 3D Context Similarity Learning

In our self-supervised 3D context similarity learning, the positive samples for a given anchor are generated by spatial and temporal augmentations while the other samples in the batch are negative. To increase negative samples, we construct a memory bank [15, 16] across batches and adopt a mining strategy to obtain the informative hard samples [19]. Considering a given anchor sample \( x_a \), only hard negative samples \( x_n \), with similarities greater than the minimum positive similarity \( (S_{pos, min} + \text{margin}) \), and hard positive samples \( x_p \), with similarities smaller than the maximal negative similarity \( (S_{neg, max} - \text{margin}) \), are taken into account by the MS loss [19]:

\[
\mathcal{L}_{MS} = \frac{1}{M} \sum_{i=1}^{M} \left[ \frac{1}{\alpha} \log[1 + \sum_{k \in \mathcal{P}_i} e^{-\alpha(S_{ik} - \lambda)}] + \frac{1}{\beta} \log[1 + \sum_{k \in \mathcal{N}_i} e^{\beta(S_{ik} - \lambda)}] \right]
\]

(2)

where \( M \) is the number of training samples, \( S_{ik} \) denotes the similarity between \( x_i \) and \( x_k \), \( \mathcal{P}_i \) and \( \mathcal{N}_i \) are the mined sets of positive clips and negative clips, respectively. \( \alpha \) and \( \beta \) are fixed hyper-parameters.

Flipped Clip Similarity Loss. The MS Loss only considers the similarities between the positive and negative samples, while the relative
positive similarity discrimination ability of the model is also essential. For example, the DSVR task [13] only takes the edited version of the query as positive, while taking the other videos containing the same scenes as negative, although these videos can be treated as positive in other tasks. Therefore, we propose the Flipped Clip Similarity Loss (FCS loss), which drives the model to distinguish the relative positive similarities by flipping clips. The inspiration comes from Vision Chirality [20], which reveals that flipped images can bring a dramatic semantic change due to the asymmetry of Vision Chirality. Therefore, a flipped positive clip \( x_{p_f} \) is regarded to be further away from the anchor clip \( x_a \) than other non-flipped positive clips \( x_p \). Namely, the similarity \( Sim_{pf} \) between \( x_{pf} \) and \( x_a \) is regarded to be smaller than \( Sim_p \), the similarity between \( x_p \) and \( x_a \).

Thus, the FCS loss can be designed as:

\[
\mathcal{L}_{FCS} = \frac{1}{N} \sum \max \{ D(x_{a1}, x_{p1}) - D(x_{a1}, x_{pf1}) + \gamma, 0 \} \quad (3)
\]

where \( N \) is the batch size, \( \gamma \) is the minimum margin between the non-flipped positive pair and flipped positive pair, \( D(\cdot) \) means the distance between a pair of clips. Here we use Cosine distance.

In this way, the three kinds of similarity as discussed in [19] are all considered, while the self-similarities of different positive samples can be distinguished as well with the help of FCS loss. The whole loss is constructed by the two parts with weights \( w_1 \) and \( w_2 \):

\[
\mathcal{L} = w_1 \mathcal{L}_{MS} + w_2 \mathcal{L}_{FCS} \quad (4)
\]

**ShotMix.** Different from images, the information volume in a video varies with the duration and the shot number. When a clip contains more shots, information changes more rapidly among frames so these frames are generally very different while the frames of a clip with less shots are similar on the contrary. In similarity learning, a pair of positive clips sampled from a video containing few shots may be easy samples since their frames are very similar. Therefore, we propose ShotMix to reconstruct such easy samples by increasing random perturbations of the shot number. Formally, given a video \( V \), two frame sequences are sampled as a pair of positive clips \( x_a = \{ f_{i1} \}_{i=1}^P, x_p = \{ f_{i2} \}_{i=1}^P \) with an overlapping sub-clip. The length of the overlapped sub-clip is \( \tau * P \) and the ratio \( \tau \) randomly ranges in \((0.7, 1.0)\). Then, we further sample a cut \( x_c \) with a length ranging in \((0, (1 - \tau) * P)\) from \( V \) to replace the beginning or the end of \( x_p \) to get \( x'_p \), so that more shots from \( x_c \) can be mixed into it. When \( x_c \) is dissimilar with \( x_a, x'_p \) will get harder due to the mixed noise. Otherwise, even though \( x_c \) is similar with \( x_a \), it makes \( x'_p \) a new clip of \( V \) with a new shot order, which is harder than \( x_p \) as well.

**3.2. Predictive Masked Autoencoder (PredMAE)**

The 3D pipeline we propose is compact and efficient but brings a challenge for training due to more parameters. We consider some unsupervised representation learning methods to promote downstream tasks, such as the recent VideoMAE [23] reconstructing the randomly masked spatiotemporal video tubes as the pretext task. When the model \( f(\cdot) \) reconstructs the masked video tubes \( Q_{t+1} \), the other correlated and not masked tubes in \( \{ Q_1, \cdots, Q_{F, S} \} \) provide information depends on their distances from \( Q_{t+1} \). \( F \) is the frame number, and \( S \) is the spatial patch number in one frame. However, the distance \( < Q_{t+1}, Q_{t+2} > \) and \( < Q_{t+1, s}, Q_{t+3} > \) are the same while the direction relationship cannot be represented inadequately, which causes a lack of description about the direction of time elapsing and some motions with direction features are hard to be distinguished, such as “standing up” and “sitting down”.

Thus, we propose the PredMAE to replace the reconstruction task with future frame prediction. We randomly mask 90% spatiotemporal tubes from clip input \( X = \{ x_1, x_2, \cdots, x_t \} \) following VideoMAE [23] and get the masked clip input \( X_m = \{ x_{m1}, x_{m2}, \cdots, x_{mt} \} \). The next \( t \) frames \( X_{GT} = \{ x_{t+1}, x_{t+2}, \cdots, x_{t+2t} \} \) are the \( GT \) for future frames prediction. The prediction model is composed of an encoder \( E(\cdot) \) and a decoder \( D(\cdot) \) as shown in Fig.2. We take ViT-base/small and the divided space-time attention (namely TimeStomer [24]) followed with an Multilayer Perceptron (MLP) as the backbone of \( E(\cdot) \). \( D(\cdot) \) is a 4-layer transformer with an MLP followed. Mean Square Error (MSE) loss is adopted to constrain the prediction. Without reference from future frames, the model tends to learn the longer-range dependencies as supplement. Modeling long-range dependencies is the strength of transformer and PredMAE brings out its potential.

**4. EXPERIMENTS**

**4.1. Experiment Setting**

**Dataset.** We use Kinetics400 [26] as training dataset to train the self-supervised model without using any labels. It consists of 240k videos from Youtube. For evaluation, we test the NDVR task on CC_WEB_VIDEO [27] and FIVR-200K [13] dataset. CC_WEB_VIDEO consists of 24 query sets and 13129 videos. FIVR-200K dataset has 100 queries and 225960 videos and includes three different fine-grained retrieval tasks.

**Training Details.** In the pretraining stage, we use 8 frames as input to predict the next 8 frames. The models are trained from scratch with a mask ratio of 0.9. In the similarity learning stage, we randomly apply the base augmentation of cropping, rotation, and color-jitter. The values of \( w_1, w_2, \) and \( \gamma \) in FCS Loss are 1, 0.01, and 0.1. We use the same optimizer settings on two stages with 8 A100 GPUs. We adopt AdamW [28] as optimizer and cosine annealing as learning rate scheduler. The learning rate is adjusted by \( lr = \frac{base \cdot lr \cdot \text{batch size}}{256} \) and \( base \cdot lr \) is set to 4.

**Model Inference.** During the inference stage, we sample a clip of 8 frames from video using a temporal stride of 1 second. We test two configurations, the "Base" and the "Small" model, which used different sizes backbones. "Base": the default version of our method with 768 hidden dimensions in all Transformer Blocks. "Small": the lightweight version with 384 hidden dimensions.

**4.2. Ablation Study**

**(1) Effectiveness of Proposed Methods.** In this section, we evaluate the retrieval performance of 3D-CSL-Base. We use a randomly selected subset of K400 with 100k videos as the training dataset and FIVR-200K as the evaluation dataset. We begin with a solid baseline model initialized with ImageNet [24] and optimized with MS-Loss [19]. We assess the effectiveness of PredMAE by comparing it with two pretraining methods. To compare with VideoMAE [23], which is similar to PredMAE, we reimplement it and replace the backbone with TimeStomer. Next, we compare the performance of with/without the proposed ShotMix, FCS Loss and TopK-CS.

As shown in Table 1, our 3D-CSL outperforms the baseline by a large margin of 9%. We observe that integrating PredMAE, ShotMix, FCS Loss, and TopK-CS each contributes to improve performance. Our results suggest that the gain comes from the good pre-

| Pretrain        | SH | FL | TC | DSVR | CSVR | ISVR |
|-----------------|----|----|----|------|------|------|
| ImageNet        | -  | -  | -  | 0.773| 0.729| 0.603|
| VideoMAE [23]  | -  | -  | -  | 0.796| 0.752| 0.624|
| PredMAE (Ours)  | -  | -  | -  | 0.810| 0.760| 0.635|

Table 1. Effectiveness of the proposed methods on FIVR-200K. SH, FL and TC stand for ShotMix, FCS loss and TopK-CS respectively.

Authorized licensed use limited to the terms of the applicable license agreement with IEEE. Restrictions apply.
Table 2. Comparison of retrieval performance with SOTA on FIVR-200K and CC_WEB_VIDEO. In conjunction with Fig1, clip-level methods offer a superior trade-off between accuracy and computational/storage cost. †The dataset is collected by author and not publicly available.

(2) Efficiency of Data and Model. The robustness of our 3D-CSL allows it to work well on small datasets and small models. We validate this property on the 3D-CSL-Small and a 100k subset of K400. Compared to other datasets, we notice that K400 has the shortest duration of 635 hours, which is 31% of VCDB [14] and 21% of Youku† [12]. When aligned the video number with VCDB, the duration decreases to 265 hours. From Table2, we observe that the 3D-CSL-Small model is still competitive and achieves 0.855 on the DSVR task with only 50% parameters compared to 3D-CSL-Base.

Moreover, our models can benefit from the increased data volume and larger model size. From Table2, our 3D-CSL-Base trained on the entire 200k dataset achieves an average improvement of 2.4% on FIVR-200K. We set this model as default model to compare against recent state-of-the-art models.

4.3. Comparison to the State-of-the-Art

We compare the performance of 3D-CSL with several state-of-the-art approaches on FIVR-200K [13] and CC_WEB_VIDEO [27]. We report the best results of the original papers. For video-level methods, we report DML [5] and TCA-C [2]. For frame-level methods, we report TN [29], PPT [30], TCA-F [2], Visil-V [1], and VRL-F [12]. As for clip-level methods, we have found only one related work (VRL [12]), which is also proposed recently. It is noted that VRL still uses a 2D model for frame-by-frame feature extraction and an aggregation layer to get the clip features. However, our 3D-CSL extract clip features through the entire network at once.

(1) Retrieval Performance. As Table2 shows, our 3D-CSL achieves the best performance on clip-level methods. Even though we do not use complex editing transformations like VRL, our models trained on base transformation and ShotMix still learn the robustness representation. Compared with frame-level methods on FIVR-200K, our 3D-CSL outperforms all of them except Visil and VRL-F on DSVR and ISVR tasks. We conjecture that these mAPs are decreased due to the 8 times higher compression rate of the clip-level features, which aggregates 8 frames information into one embedding.

Furthermore, on the most challenging ISVR task, 3D-CSL-Base can achieve 0.711 mAP, outperforming all state-of-the-art methods. In this task, some reference videos neither temporally nor spatially overlap with the queries, which greatly increases the retrieval difficulty. The superior result indicates that our model has a powerful ability for long-range spatiotemporal modeling.

(2) Retrieval Speed and Storage. Comprehensive performance comparison on retrieval speed and storage is shown in Fig1. All models are tested on Nvidia 2080Ti. The proposed method achieves the best trade-off between efficiency and effectiveness. The video-level methods require minimal feature storage and retrieval cost, but their low accuracy is insufficient for large-scale accurate retrieval. Compared with these methods, our method achieves a significant improvement by 30.9%, 28.2%, and 23.8% mAPs on FIVR-200K from Table2. Compared with the frame-level methods, which require $O(NM)$ complexities for each pair in Eq1, our clip-level method performs only $O(NM/64)$ complexities per pair. Besides, the size of space for storing features is also decreased by 8 times.

4.4. Visualization of Learned Features

Our compact pipeline enables us to explore the inherent learning pattern of similarity learning. For better understanding, we compare it with the supervised action classification task [24]. By visualizing the attention flow of learned features with [31], we observe a significant difference between the two tasks. As Fig3 shows, the classification features focus more on relevant objects for reasoning while ignoring background. In contrast, the similarity features have wider attention spans, which is helpful for the model to distinguish the fine-grained details between videos.

Fig. 3. Visualization results on Kinetics400. (Top) the input clip; (Middle) attention of classification features; (Bottom) attention of similarity features.

5. CONCLUSION

This paper introduces 3D-CSL, a new pipeline for modeling clip-level features for the NDVR task. It shows potential for long-range video representation learning with speed & storage advantages in NDVR. Although the clip-level features are more compact than frame features, they can achieve state-of-the-art results in ISVR trained with the proposed self-supervised training strategy. The proposed FCS loss enables the model to distinguish the relative magnitudes of positive similarities, and the video-specific ShotMix strengthens the model’s robustness facing multi-shot videos. Furthermore, the proposed PredMAE can boost the 3D pipeline significantly for pretraining it with future frame prediction as a pretext task, which encourages the model to learn the long-range dependencies in videos efficiently.
6. REFERENCES

[1] Giorgos Kordopatis-Zilos, Symeon Papadopoulos, Ioannis Patras, and Ioannis Kompatiaris, “Visil: Fine-grained spatiotemporal video similarity learning,” in Proceedings of the IEEE/CVF International Conference on Computer Vision, 2019, pp. 6351–6360.

[2] 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, 2021, pp. 3268–3278.

[3] Hao Liu, Qingjie Zhao, Hao Wang, Peng Lv, and Yanning Chen, “An image-based near-duplicate video retrieval and localization using improved edit distance,” Multimedia Tools and Applications, vol. 76, no. 22, pp. 24435–24456, 2017.

[4] Chien-Li Chou, Hua-Tsung Chen, and Suh-Yin Lee, “Pattern-based near-duplicate video retrieval and localization on web-scaled videos,” IEEE Transactions on Multimedia, vol. 17, no. 3, pp. 382–395, 2015.

[5] G. Kordopatis-Zilos, S. Papadopoulos, I. Patras, and Y. Kompatiaris, “Near-duplicate video retrieval with deep metric learning,” in Proceedings of the IEEE international conference on computer vision workshops, 2017, pp. 347–356.

[6] H. Lee, J. Lee, J.Y.H. Ng, and P. Natsev, “Large scale video representation learning via relational graph clustering,” in Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2020, pp. 6807–6816.

[7] J. Song, Y. Yang, Z. Huang, H. T. Shen, and R. Hong, “Multiple feature hashing for real-time large scale near-duplicate video retrieval,” in Proceedings of the 19th ACM international conference on Multimedia, 2011, pp. 423–432.

[8] 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, vol. 27, no. 7, pp. 3210–3221, 2018.

[9] Li Yuan, Tao Wang, Xiaopeng Zhang, Francis EH Tay, Zequn Jie, Wei Liu, and Jashii Feng, “Central similarity quantization for efficient image and video retrieval,” in Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, 2020, pp. 3083–3092.

[10] Giorgos Kordopatis-Zilos, Symeon Papadopoulos, Ioannis Patras, and Yiannis Kompatiaris, “Near-duplicate video retrieval by aggregating intermediate cnn layers,” in International conference on multimedia modeling. Springer, 2017, pp. 251–263.

[11] Siying Liang and Ping Wang, “An efficient hierarchical near-duplicate video detection algorithm based on deep semantic features,” in International Conference on Multimedia Modeling. Springer, 2020, pp. 752–763.

[12] Xiangteng He, Yulin Pan, Mingqian Tang, Yiliang Lv, and Yuxin Peng, “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, 2022, pp. 1002–1011.

[13] Giorgos Kordopatis-Zilos, Symeon Papadopoulos, Ioannis Patras, and Ioannis Kompatiaris, “Fivr: Fine-grained incident video retrieval,” IEEE Transactions on Multimedia, vol. 21, no. 10, pp. 2638–2652, 2019.

[14] Y.G. Jiang, Y. Jiang, and J. Wang, “Vcdb: a large-scale database for partial copy detection in videos,” in European conference on computer vision. Springer, 2014, pp. 357–371.

[15] K. He, H. Fan, Y. Wu, S. Xie, and R. Girshick, “Momentum contrast for unsupervised visual representation learning,” in Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, 2020, pp. 9729–9738.

[16] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton, “A simple framework for contrastive learning of visual representations,” in International conference on machine learning. PMLR, 2020, pp. 1597–1607.

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

[18] Yifan Sun, Changmao Cheng, Yuhua Zhang, Chi Zhang, Liang Zheng, Zhongdao Wang, and Yichen Wei, “Circle loss: A unified perspective of pair similarity optimization,” in Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2020, pp. 6398–6407.

[19] Xin Wang, Xintong Han, Weilin Huang, Dengke Dong, and Matthew R. Scott, “Multi-similarity loss with general pair weighting for deep metric learning,” in Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2019, pp. 5022–5030.

[20] Z. Lin, J. Sun, A. Davis, and N. Snavely, “Visual chirality,” in Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2020, pp. 12295–12303.

[21] B. Fernando, H. Bilen, E. Gavves, and S. Gould, “Self-supervised video representation learning with odd-one-out networks,” in Proceedings of the IEEE conference on computer vision and pattern recognition, 2017, pp. 3636–3645.

[22] Dejing Xu, Jun Xiao, Zhao Zhao, Jian Shao, Di Xie, and Yueting Zhuang, “Self-supervised spatiotemporal learning via video clip order prediction,” in Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2019, pp. 10334–10343.

[23] Zhan Tong, Yibing Song, Jue Wang, and Limin Wang, “Videomae: Masked autoencoders are data-efficient learners for self-supervised video pre-training,” arXiv preprint arXiv:2103.12602, 2022.

[24] Gedas Bertasius, Heng Wang, and Lorenzo Torresani, “Is space-time attention all you need for video understanding?” in ICML, 2021, vol. 2, p. 4.

[25] Satya Krishna Gorni, Noël Voutissis, Junwei Ma, Keyvan Golestan, Maksims Volkovs, Animesh Garg, and Guangwei Yu, “X-pool: Cross-modal language-video attention for text-video retrieval,” in Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2022, pp. 5006–5015.

[26] Joao Carreira and Andrew Zisserman, “Quo vadis, action recognition? a new model and the kinetics dataset,” in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2017, pp. 6299–6308.

[27] 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, 2007, pp. 218–227.

[28] Ilya Loshchilov and Frank Hutter, “Decoupled weight decay regularization,” arXiv preprint arXiv:1711.05101, 2017.

[29] H.K. Tan, C.W. Ngo, R. Hong, and T.S. Chua, “Scalable detection of partial near-duplicate videos by visual-temporal consistency,” in Proceedings of the 17th ACM international conference on Multimedia, 2009, pp. 145–154.

[30] 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, vol. 17, no. 3, pp. 382–395, 2015.

[31] Samira Abmar and Willem Zuidema, “Quantifying attention flow in transformers,” arXiv preprint arXiv:2005.00928, 2020.