Efficient Cross-Modal Video Retrieval With Meta-Optimized Frames

Ning Han , Xun Yang , Ee-Peng Lim , Hao Chen , and Qianru Sun , Member, IEEE

Abstract—Cross-modal video retrieval aims to retrieve semantically relevant videos when given a textual query, and is one of the fundamental multimedia tasks. Most top-performing methods primarily leverage Vision Transformer (ViT) to extract video features (Lei et al., 2021), (Bain et al., 2021), (Wang et al., 2022). However, they suffer from the high computational complexity of ViT, especially when encoding long videos. A common and simple solution is to uniformly sample a small number (e.g., 4 or 8) of frames from the target video (instead of using the whole video) as ViT inputs. The number of frames has a strong influence on the performance of ViT; e.g., using 8 frames yields better performance than using 4 frames but requires more computational resources, resulting in a trade-off. To get free from this trade-off, this paper introduces an automatic video compression method based on a bilevel optimization program (BOP) consisting of both model-level (i.e., base-level) and frame-level (i.e., meta-level) optimizations. The model-level optimization process learns a cross-modal video retrieval model whose input includes the “compressed frames” learned by frame-level optimization. In turn, frame-level optimization is achieved through gradient descent using the meta loss of the video retrieval model computed on the whole video. We call this BOP method (as well as the “compressed frames”) the Meta-Optimized Frames (MOF) approach. By incorporating MOF, the video retrieval model is able to utilize the information of whole videos (for training) while taking only a small number of input frames in its actual implementation. The convergence of MOF is guaranteed by meta gradient descent algorithms. For evaluation purposes, we conduct extensive cross-modal video retrieval experiments on three large-scale benchmarks: MSR-VTT, MSVD, and DiDeMo. Our results show that MOF is a generic and efficient method that boost multiple baseline methods, and can achieve a new state-of-the-art performance.

Index Terms—Cross-modal, multimodal, video retrieval, video compression.

I. INTRODUCTION

The exponential growth of video data on the Internet has attracted much attention to video recognition [5], [6], [7], [8] and video retrieval tasks [9], [10], [11], [12], [13], [14]. The cross-modal video retrieval task takes a text as a query and requires the utilized model to search for the most relevant videos based on this query. The model usually contains two neural network modules—one for text encoding and one for video encoding [2], [3]. The video encoding module is expected to learn spatial-temporal patterns in the input video that can be aligned with the semantics in the query text. The state-of-the-art video encoder is based on Vision Transformer (ViT) [15], [16], [17], and is trained with a text encoding module in an end-to-end manner [18], [19]. Nevertheless, a ViT is costly when encoding long videos. Its complexity grows quadratically along with the number of input frames [16]. For example, when addressing a video with N frames, applying a ViT on flattened spatio-temporal token sequences (where the length of the sequence is S for each frame) introduces \( O(N^2S^2) \) exponential complexity for the both training and inference processes.

A simple and common solution is to uniformly sample a handful of frames as ViT inputs, instead of using the original whole video. However, the “number of sampled frame” is a hyperparameter that must be carefully selected. In addition, the use of sparse sampling leads to a trade-off between effectiveness (i.e., retrieval accuracy) and efficiency (i.e., computing speed) when using ViT-based video retrieval models.

We consider a few examples involving conventional methods, which sample frames as the inputs of ViT. Bain et al. [2] used a curriculum learning approach to sparsely sample frames in different training iterations. They limited the number of frames sampled per video to only 4 or 8 and these sampled frames were passed to the ViT in each iteration. However, this sampling method with a fixed number of frames is sub-optimal and does not adapt to videos with different difficulty levels. For example, in Fig. 1, we show three video (and query) examples ordered from easy to hard. For the example in Fig. 1(a), any single input frame from the video is sufficient for making a correct relevance prediction (for the model). For the example in Fig. 1(b), a single input frame from the video is not sufficient for the model; i.e., the ground-truth video is ranked 6th. We now need all 8 input frames sampled from the video. Finally, for the example in Fig. 1(c), even all 8 input frames are still not sufficient for the model to retrieve the ground-truth video as it is ranked 7th. We now need all 12 input frames sampled from the video.

Manuscript received 6 September 2022; revised 6 February 2023 and 10 July 2023; accepted 7 June 2024. Date of publication 28 June 2024; date of current version 21 November 2024. This work was supported in part by the National Natural Science Foundation of China under Grant U22A2094, in part by A*STAR under its AME YIRG Grant A20E6c0101, and in part by the Natural Science Foundation of Hunan Province under Grant 2024J16343. The associate editor coordinating the review of this article and approving it for publication was Dr. Xi Li. (Corresponding authors: Xun Yang; Qianru Sun.)

Ning Han is with the School of Computer Science, Xiangtan University, Xiangtan 411105, China (e-mail: ninghan@xtu.edu.cn).

Xun Yang is with the School of Information Science and Technology, University of Science and Technology of China, Hefei 230026, China (e-mail: xyang21@ustc.edu.cn).

Ee-Peng Lim and Qianru Sun are with the School of Computing and Information Systems, Singapore Management University, Singapore 178902 (e-mail: eplim@smu.edu.sg; qianrusun@smu.edu.sg).

Hao Chen is with the Department of Information Science and Engineering, Hunan University, Changsha 410082, China (e-mail: chenhao@hnu.edu.cn).

Our code is publicly available at: https://github.com/lionel-hing/MOF.

Digital Object Identifier 10.1109/TMM.2024.3416669
Fig. 1. Examples of easy (a), moderate (b) and hard (c) text-video pairs on the MSR-VTT 1k-A test set [4]. Showcasing the ranking variation of ground-truth videos with different numbers of sampled frames retrieved by text queries. The red bounding box denotes the numbers of sampled frames used for training a video retrieval model (e.g., Frozen [2]) and testing. The ranking of ground-truth video is output by the trained model. It is intuitive that easier (harder) samples need a smaller (larger) number of input frames. More frames may be needed for extremely hard samples, e.g., that in (c).

To quantify the effect of the number of frames on model performance, we train our model using Frozen [2] (which is a top-performing method and is taken as our baseline) with varying numbers of frames and report the resulting text-video retrieval performance and query test time in Fig. 2. In particular, Fig. 2 shows that training with a larger number of frames achieves better model performance but requires a significantly longer test time. Existing methods [16], [21], [22] for reducing this inference time cost are mostly focused on selecting a few informative tokens or reducing the complexity of quadratic attention which unfortunately offers limited efficiency.

In contrast, our idea is to meta-learn the most representative patterns from the whole input video in an adaptive and automatic method. These patterns are fed into the ViT under a strict token length constraint so that, they do not increase any test time. Our method is based on bilevel optimization (i.e., meta-learning). Both the method and its resulting frames are called Meta-Optimized Frames (MOF). By incorporating MOF, video retrieval models are able to utilize the whole given video for cross-modal inference without increasing the number of input frames required by the ViT. As demonstrated in Fig. 2, utilizing MOF as a plug-in module improves the performance of the baseline method (Frozen [2]) under different input frame settings.

Technically for MOF, we perform optimization at two levels: the model-level (i.e., the base-level in conventional meta-learning or bilevel optimization [23], [24]) and the frame-level (i.e., the meta-level in conventional meta-learning or bilevel optimization [23], [24]). In our case, model-level optimization takes the result of frame-level optimization (i.e., MOF) as the ViT input and aims to train the network parameters of ViT. Frame-level optimization takes the ViT input as “parameters”, i.e., a selected number of video frames are parameterized to be learnable with gradient descent, and optimizes them (i.e., with MOF) using the meta loss of the ViT computed on the whole video. We formulate these two levels in a bilevel optimization program (BOP) [23], [24] that alternates between learning at the two optimization levels, throughout the entire training phase. Each alternation is called a BOP phase. In each phase, we execute a local BOP to distill the knowledge contained in regular frames into the MOF. Specifically, we first train a temporary model by taking MOF as the input layer, where MOF in the beginning is initialized with uniformly sampled frames. Then, we compute a validation loss with the whole video as input, and use the derived gradients to back-propagate and optimize the input layer (during training), i.e., the parameters of the meta-optimized frames. We conduct extensive experiments to evaluate the proposed MOF as a plug-in module in state-of-the-art methods, such as Frozen [2], and CLIP4clip [25]. Our results obtained on three cross-modal retrieval benchmarks show that 2.2%, 2.1%, and 1.9% R@10 improvements are achieved over Frozen [2], demonstrating the effectiveness of MOF.

Our contributions are thus three-fold: (1) a novel meta-learning method called MOF to adaptively and automatically compress a video into fewer but more representative frames;
(2) a novel BOP-based formulation and an end-to-end training solution that alternates between the learning of MOF and cross-modal video retrieval models; and (3) extensive experiments conduct on three standard benchmarks to validate the effectiveness of MOF.

The rest of this paper is organized as follows. Section II reviews the related works. Section III-A introduces the preliminaries of this video retrieval task. Section III shows the details of our proposed MOF method. Section V presents extensive experiments and discussions concerning the results. Finally, Section VI summarizes a few conclusions.

II. RELATED WORK

In this section, we first briefly introduce the the progress of cross-modal video retrieval. Then, we review the most related works from two aspects. The first aspect involves a bilevel optimization program, the second concerns representative frame detection, and the final one addresses efficient cross-modal retrieval.

A. Cross-Modal Video Retrieval

The core problem of cross-modal video retrieval is to measure the similarity between different modal features. A typical solution is to construct a common embedding space to learn video-textual correlations. Based on the training approach, we can divide existing works into two groups: video feature-based non-end-to-end and raw video-based end-to-end methods.

Video feature-based non-end-to-end methods: [26], [27], [28], [29], [30], [31], [32] are mainly based on video features generated via off-the-shelf video feature extractors as vision embedding and fed into the joint embedding space along with text to measure similarity. For example, Dong et al. [26], [33] adopt three branches, i.e. mean pooling, bi-GRU, and CNN to encode sequential videos and texts and learn a hybrid common space for video-text retrieval. Some studies have also explored rich multimodal information (e.g., motion, audio, and speech) [34], [35], [36], [37] or large-scale datasets (e.g., HowTo100M) pre-training [38], [39], [40] to improve the performance of cross-modal retrieval. Chen et al. and Han et al. [27], [28] propose fine-grained alignment models that decompose text and video into multiple levels and align text with video at multiple levels for video-text matching.

Raw video-based end-to-end methods: [1], [2], [3] train the model with raw video and paired text in an end-to-end manner. For instance, Lei et al. [1] propose a general framework ClipBERT that enables end-to-end pre-training through a sparse sampling strategy. Bain et al. [2] adopt a transformer-based video backbone and design a curriculum learning schedule to train the model on both image and video datasets to further improve performance. Inspired by Frozen [2], Wang et al. [3] present a transformer-based object-aware dual encoder model for end-to-end video-language pre-training. However, they both face the potential problem of reducing the high computational overload of dense video inputs.

Recently, pre-trained CLIP-based methods [22], [25], [41], [42], [43], [44] have achieved superior results, thanks to the direct transfer of powerful knowledge from the pre-trained CLIP [45] and continued pre-training on a large-scale video-language dataset. The transfer of CLIP is approached from diverse viewpoints, encompassing feature aggregation [25], [41], [44] and semantic alignment [22], [43], [46]. Different from these works, our work aims to achieve a decent trade-off between accuracy and speed by meta-learning efficient input frames. Our paradigm can therefore accommodate pre-trained CLIP-based and transformer-based object-aware models [3], [25]. Both pre-training CLIP-based and transformer-based object-aware models can be directly integrated into our training framework (e.g., as backbones) to achieve additional performance gain.

B. Bilevel Optimization Program

Bilevel optimization program (BOP) [47], [48], [49] aims to solve two levels of problems in one framework where an optimization problem contains another optimization problem as a constraint. A variety of problems arising in the area of machine learning can be formulated by bilevel optimization programs (BOP): incremental learning, adversarial training, and meta-learning. Recently, Liu et al. [50] propose a bilevel optimization-based approach for tackling the stability-plasticity dilemma in class incremental learning tasks. Zhu et al. [51] re-design vanilla GAN [48] can be formulated as a BOP, which maximizes the reality score of generated images and minimizes the real-fake reconstruct loss. Wei et al. [52] present a meta self-paced network that automatically learns a weighting scheme from data for cross-modal matching. It is known by [53] that hyperparameter optimization [49], [54], [55] is related to meta-learning. Therefore, if taken the image input layer as a hyperparameter layer in the cross-modal video retrieval network, our optimization method is related to meta-learning [49], [50], [52]. Meta-learning [54] usually takes a bilevel optimization process: base-level and meta-level (and in our case they are respectively called model-level and frame-level), where the base-level is to learn the final model on the support set, and meta-level is to optimize the hyperparameters on the query/validation set. In our work, we design a new version of BOP that alternatively optimizes the parameters of the retrieval models and the MOF across entire training phase. Specifically, we design the process of compressing a big validation set (e.g., 32 frames) to a small support set (e.g., 8 frames) to realize the purpose of efficient learning with limited frames. This process is called MOF and it “learns to compress” the video into fewer but more representative frames. In each iteration, we apply a local BOP to learn the MOF specific to the input whole video.

C. Representative Frame Detection

Detecting representative frames from video input has been validated to be useful in video recognition and retrieval tasks [56], [57]. To be specific, Mithun et al. [56] used the key frames extracted by a dissimilarity-based sparse subset selection approach and evaluated it in the video-text retrieval task. Our work is also focused on learning representative frames for video retrieval tasks, and our key difference with [56] is: [56] extract keyframes only for video using a dissimilarity-based sparse
subset selection approach [58], while ours is a gradient descent based method in a bilevel optimization program which is built on the top of the base model (cross-modal model) to naturally incorporates text inputs. Bilen et al. [59] proposed to generate a dynamic image for each video by a rank pooling technique that captures the temporal evolution of actions. Its difference from ours is that it transforms the video to a single image with a manually designed method but it is not learnable (ours is learnable and thus optimal).

More recently, Tavakolian et al. [60] also proposed to generate a single image from the video, and their method is based on adversarial learning. Qiu et al. [57] proposed to learn the transformation from the video to an informative frame and applied it to the task of video recognition. Our work is different from these two works mainly in two folds: 1) our method is flexible to generate any number (not only one) of images with a simple and elegant gradient descent process; and 2) our method works in an end-to-end manner with the “participation” of text inputs in the optimization process, that learns better “frames” for cross-modal retrieval tasks, i.e., between images and texts.

D. Efficient Cross-Modal Retrieval

Recently, ViT-based architectures have been adapted to accommodate cross-modal video retrieval tasks [2], [22], [25], [41], [42]. Some works [2], [22] reduce the quadratic attention complexity and select a few informative tokens to make ViT-based architectures more efficient for cross-modal video retrieval tasks, which is complementary to our approach. For example, Bain et al. [2] modify a ViT-based video architecture with divided space and time attention to reduce the quadratic attention complexity. To achieve a decent reduction in memory cost and speed up the inference process, Zhao et al. [22] design a multi-segment token clustering algorithm to find the most representative tokens and drop the non-essential ones for cross-modal video retrieval. Furthermore, some studies [61], [62], [63], [64] have also explored cross-modality sharable representation to improve the efficiency of cross-modal retrieval. Chen et al. [61] propose heterogeneous graph embeddings to preserve more abundant cross-modal information. Hu et al. [64] develop a cross-modal hashing module to project these two heterogeneous representations into a shared isomorphic Hamming space for compact hash code learning and boosts the efficiency of moment localization. Lu et al. [62] design an efficient online hashing module to simultaneously correlate the learned hash codes with low-level data distribution and high-level semantics, achieving high learning efficiency and retrieval accuracy. Different from these mentioned methods, our method aims to learn the representative information from a larger number of frames and compress it into fewer frames, that learns representative “frames” to reduce the computation cost of inference for cross-modal video retrieval.

III. METHODOLOGY

In this section, we first provide the preliminaries required for cross-model retrieval. We then introduce the proposed meta-optimized frames (MOP) approach in detail, including the global BOP, the model-level optimization process, the frame-level optimization procedure, and the algorithmic flow.

A. Preliminaries

End-to-end video retrieval is proposed in [1], [2] to learn joint representations directly from video frame pixels and raw text tokens instead of offline-extracted single-modality features. We use \( P = (V, T) \) to denote a video-text pair, where \( V \in \mathbb{R}^{B \times C \times W \times H} \) is a video, and \( T \) is its associated textual description; \( C \) denotes the color channels, and \( (H, W) \) is the resolution of each raw frame. For a video clip \( V \), a typical approach is to uniformly sample a sequence of video frames \( \{v_1, \ldots, v_M\} \) from \( V \) with a prespecified temporal video resolution, where \( M \) is the length of the sampled video sequence. The prevailing cross-modal video retrieval model consists of a text encoder \( F_t \) and a video encoder \( F_v \). Given a text \( T \) and a video \( V \), \( F_t \) and \( F_v \) encode their video and textual descriptions as \( E^v \) and \( E^t \), respectively, before mapping them to a joint embedding space, where the video-text similarity can be directly measured. With some similarity metrics, the encoded representations in \( E^v \) and \( E^t \) should be close if they are related; otherwise, they are far apart. Following [2], we employ a symmetrical contrastive loss [65] for cross-modal video retrieval.

The video-to-text loss \( \mathcal{L}_{v\rightarrow t} \) and text-to-video loss \( \mathcal{L}_{t\rightarrow v} \) can then be formulated as:

\[
\mathcal{L}_{v\rightarrow t} = - \frac{1}{B} \sum_{i}^{B} \log \frac{\exp(\langle E^t_i, E^v_i \rangle / \sigma)}{\sum_{j=1}^{B} \exp(\langle E^t_i, E^v_j \rangle / \sigma)},
\]

\[
\mathcal{L}_{t\rightarrow v} = - \frac{1}{B} \sum_{i}^{B} \log \frac{\exp(\langle E^v_i, E^t_i \rangle / \sigma)}{\sum_{j=1}^{B} \exp(\langle E^v_i, E^t_j \rangle / \sigma)},
\]

where \( E^v_i \) and \( E^t_i \) are the normalized embeddings of the \( i \)-th video and the \( j \)-th text, respectively, in a batch of size \( B \), and \( \sigma \) is the temperature. The overall loss function for training the retrieval model is the sum of the video-to-text loss (\( \mathcal{L}_{v\rightarrow t} \)) and the text-to-video loss (\( \mathcal{L}_{t\rightarrow v} \)):

\[
\mathcal{L}_c = \mathcal{L}_{v\rightarrow t} + \mathcal{L}_{t\rightarrow v}.
\]

Recent breakthroughs in end-to-end training strategies with a single or a few sparsely sampled frames (or clips) indicate that the appropriate sparse sampling strategy is the key to facilitating good and fast learning. In this paper, we aim to tackle the cross-modal video retrieval problem by automatically compressing a video into fewer but more representative frames. Specifically, we formulate this alternative learning strategy with a new bilevel optimization program (BOP) design composed of both model-level and frame-level optimizations, especially for addressing cross-modal video retrieval tasks. After that, the details of the video and text encoders are further elaborated.

Video and Text Encoders: Early works typically used 2D/3D CNNs as video encoders to extract offline video features [26], [27], [31]. Recently, several works [2], [8], [16] have employed visual transformers (ViT) [15] to extract video features in an end-to-end manner, demonstrating promising video modeling
performance. The structure of a ViT is similar to that of a transformer in NLP tasks, which splits an input image into a sequence of patches and then flattens them into vectors (tokens) as inputs for transformer layers. In this work, we choose the dual-encoder Frozen framework [2] as our base model to extract both video and text features in a trainable way. For the video encoder, we adopt the ViT with spatial-temporal attention from TimeSformer [2] as the backbone. More precisely, for the ViT, we use a ViT-B/16 model [15] with 12 layers, widths of 768, and 12 attention heads. Then, the output of the [CLS] token at the final layer is extracted as the visual representation. For the text encoder, similar to previous methods [1], [2], we choose the transformer-based DistillBERT structure [19] and treat the [CLS] token of the last hidden layer as the text representation. The video and text representations are later normalized by layer normalization and linearly projected into the joint video-text embedding space.

IV. META-OPTIMIZED FRAMES (MOF)

As illustrated in Fig. 3, the proposed MOF method jointly works with a retrieval model in an alternative manner, where one model updates after the update phase of the other model. MOF is initialized with uniformly sampled frames from the input video and then optimized while keeping the retrieval model temporarily fixed. Then, MOF is fed into the retrieval model as an input to train the model parameters. We formulate this alternative learning process with a global BOP (Section IV-A) and name these two steps model-level optimization and frame-level optimization. We provide the details of the proposed solutions, i.e., the model-level and frame-level optimization processes, in Sections IV-B and IV-C, respectively. The training steps are elaborated in Section IV-D.

A. Global BOP

We can separate the global BOP into a sequence of bilevel optimization phases. In each phase, the retrieval model (denoted as $\Theta$) is optimized on MOF (denoted as $\varepsilon$), where the MOF parameters are meta-optimized by using a large set of regular frames (denoted as $V^R$) sampled from the original video. Here, $R$ denotes the number of sampled frames, which is much higher than the number of frames in $\varepsilon$. MOF is trained based on the retrieval loss of $V^R$. It thus represents the “compressed” information of $V^R$. Then, it is used as an input to train the retrieval model. During this bilevel optimization process, the optimality of the retrieval model endows a constraint for optimizing MOF, and vice versa.

Specifically, in the $i$-th BOP phase, we aim to learn $\Theta_i$ to approximate the ideal model $\Theta^*_i$ by minimizing the video retrieval loss $\mathcal{L}_c$ on the current MOF $\varepsilon_i$. We formulate this step as follows:

$$\Theta^*_i = \arg \min_{\Theta_i} \mathcal{L}_c(\Theta_i(\varepsilon_i); V^R_i),$$

(4)

where $\varepsilon_i$ and $V^R_i$ represent the meta-optimized frames (i.e., MOF) and the sampled regular frames in the $i$-th phase, respectively.

In the global BOP, MOF $\varepsilon_i$ is optimized with the corresponding regular frames $V^R_i$ throughout all phases. To learn more representative frames $\varepsilon_i$ from the regular frames $V^R_i$, our learning target for MOF is to minimize the model loss on these frames, i.e., $V^R_i$. In particular, we define the objective of the global BOP as

$$\min_{\Theta_i} \mathcal{L}_c(\Theta_i(\varepsilon_i^*); V^R_i)$$

(5)

s.t. $\varepsilon_i^* = \arg \min_{\varepsilon_{i-1}} \mathcal{L}_c(\Theta_{i-1}(\varepsilon_{i-1}); V^R_{i-1}),$

(6)

where $\Theta_{i-1}(\varepsilon_{i-1})$ denotes that the model $\Theta_{i-1}$ is trained on $\varepsilon_{i-1}$ to transfer the knowledge from the regular frames $V^R_{i-1}$ to MOF $\varepsilon_i$ during the $(i-1)$-th phase. Please note that $\varepsilon_i^*$ is then used as the input in the $i$-th phase.

B. Model-Level Optimization

Fig. 3 shows that in the $i$-th phase, we first solve the model-level optimization problem with 1) MOF $\varepsilon_i$ as part of the input and 2) the model of the previous training phase $\Theta_{i-1}$ as the initialization of model $\Theta_i$. Let $\alpha$ be the learning rate of the fine-tuned retrieval models; $\Theta_i$ can be updated with one (or a few) gradient descent step(s) as

$$\Theta^*_i \leftarrow \Theta_i - \alpha \nabla_{\Theta_i} \mathcal{L}_c(\Theta_i(\varepsilon_i)).$$

(7)

Then, $\Theta_i^*$ is temporarily fixed and deployed to learn MOF $\varepsilon_{i+1}$, i.e., to solve the frame-level optimization problem in the $i$-th phase, as elaborated in Section IV-C.
C. Frame-Level Optimization

In general, the number of frames fed into the retrieval model is considerably smaller than the total number of frames in the original video $V$. The existing methods [1], [2] assume that models trained on a few frames can achieve acceptable performance. However, this assumption cannot be guaranteed, especially when these frames are randomly or uniformly sampled. In other words, these sampled frames may not be representative or discriminative. In contrast, our approach based on optimization explicitly learns a feasible approximation for this assumption given the differentiability of our MOF algorithm and guarantees the representativeness of the "sampled" frames.

To achieve this, we initialize a temporary model $\Theta_i'$ (in the $i$-th phase) from model $\Theta_i^*$ (i.e., $\Theta_i' = \Theta_i^*$) and then train it on $\varepsilon_i$ to maximize its prediction capabilities on the regular frames $V_i^R$. We use $V_i^R$ to compute a validation loss to penalize this temporary training process with respect to the parameters of $\varepsilon_i$. The entire optimization procedure is executed within a single stage, called the local BOP, and can be formulated as

$$\min_{\varepsilon_i} \mathcal{L}_c(\Theta_i'(\varepsilon_i); V_i^R)$$

s.t. $\Theta_i'(\varepsilon_i) = \arg\min_{\Theta_i} \mathcal{L}_c(\Theta_i; \varepsilon_i).$ (8)

*Training $\varepsilon_i$: The training flow is detailed in Fig. 4. First, the parameter size of $\varepsilon_i$ (i.e., MOF) is initialized by $V_i$. Second, we initialize a temporary model $\Theta_i''$ with $\Theta_i'$ and train $\Theta_i''$ on $\varepsilon_i$ for one (or a few) iteration(s) via gradient descent as follows:

$$\Theta_i'' \leftarrow \Theta_i' - \alpha \nabla_{\Theta_i''} \mathcal{L}_c(\Theta_i'(\varepsilon_i); V_i^R),$$

where $\alpha$ is the learning rate of the fine-tuned temporary models. Finally, as both $\Theta_i'$ and $\varepsilon_i$ are differentiable, we are able to compute the loss of $\Theta_i''$ on $V_i^R$ and backpropagate this validation loss to optimize $\varepsilon_i$:

$$\varepsilon_{i+1} \leftarrow \varepsilon_i - \beta \nabla_{\varepsilon_i} \mathcal{L}_c(\Theta_i''(\varepsilon_i); V_i^R),$$

where $\beta$ is the learning rate. In this step, we need to backpropagate the validation gradients until the input layer is reached by unrolling all training gradients of $\Theta_i'$. This operation involves propagating a gradient through a gradient. Computationally, it requires an additional backward pass through $\mathcal{L}_c(\Theta_i''; \varepsilon_i)$ to compute Hessian-vector products, which is feasible by using deep learning toolboxes.

D. Algorithm

In Algorithm 1, we summarize the entire proposed MOF training process. We show the alternative training mechanism between the retrieval models and the parameters of MOF, corresponding to Sections IV-A–IV-C, respectively. Specifically, in each phase, steps 2–9 are applied for model-level training, and steps 12–16 are applied for frame-level training. Step 10 evaluates the learned model $\Theta_i$ for the current phase, and the best value among all phases is reported as the final evaluation.

Algorithm 1: Meta-Optimized Frames (MOF) Training Process.

**Input:** An untrimmed video $V$, $R$: the number of regular frames, $t$: the maximum number of phases, $\alpha$ and $\beta$: the learning rates.

**Output:** The retrieval model $\Theta_i$ and MOF $\varepsilon_i$.

1: for $i = 0, 1, 2, \ldots, t$ do
2: Obtain a real training video $V_i$;
3: if $i = 0$ then
4: Uniformly sample frames from $V_0$ to initialize $\varepsilon_0$;
5: Initialize $\Theta_0$ and train it on $\varepsilon_0$;
6: else
7: Obtain $\varepsilon_i$ from memory;
8: Initialize $\Theta_i$ with $\Theta_{i-1}$;
9: Train $\Theta_i' \leftarrow \Theta_i - \alpha \nabla_{\Theta_i} \mathcal{L}_c(\Theta_i'(\varepsilon_i));$
10: Run a test and record the results;
11: end if
12: Randomly sample frames $V_i^R$ from $V_i$;
13: Initialize $\Theta_i''$ with $\Theta_i'$;
14: Train $\Theta_i'' \leftarrow \Theta_i' - \alpha \nabla_{\Theta_i''} \mathcal{L}_c(\Theta_i''(\varepsilon_i); V_i^R)$;
15: Train $\varepsilon_{i+1} \leftarrow \varepsilon_i - \beta \nabla_{\varepsilon_i} \mathcal{L}_c(\Theta_i''(\varepsilon_i); V_i^R)$;
16: Update $\varepsilon_{i+1}$ in memory.
17: end for
DiDeMo [67] contains 10,000 videos annotated with 40,000 sentences. Following [1], [2], [35], we evaluate video-paragraph retrieval, where all sentence descriptions for a video are concatenated into a single query.

B. Evaluation Metrics

We adopt the widely used median rank (MedR) and recall rate at top K (R@K) for assessing retrieval accuracy. MedR calculates the median rank position among where true positives are returned. R@K measures the fraction of true positives being ranked at the top K returned results. The higher R@K and lower MedR indicate better performance.

C. Implementation Details

1) The Architectures of $\Theta$: Following Frozen [2], we use Timesformer [16] and multilayer bidirectional transformer as visual encoder and text encoder, respectively. For initialization of the model weights, we initialize the entire model with Frozen weights trained on the Google Conceptual Headings (CC3M) [69] dataset and the WebVid-2M dataset [2].

2) The Architectures of $\varepsilon$: It depends on the size of the video frame and the number of video frames we need. For the three datasets, we adopt video frame size 224 with patch size 16. The number of frames is set in two ways. (1) The MOF is uniformly sampled from each video and are denoted by U. Therefore, the parameter size of the frames per video is equal to $H \times W \times U$, where $H$, $W$, and $C$ represent the height, width, and channel, respectively. (2) Regular frames are randomly sampled for each video, and are denoted by R. Similarly, the parameter size of the frames per video is equal to $H \times W \times R$. Notably, we use MOF and regular frames for training, while using only regular frames at test time for efficient retrieval.

3) Hyperparameters and Configuration: We implement our proposed method using PyTorch$^1$ on 8 Tesla A100 GPUs (40G). For model-level hyperparameters, the model $\Theta_i$ is fine-tuned with AdamW optimizer. In each (i.e., i-th) phase, the learning rate $\epsilon_i$ is initialized as 1e-5. In 1 and 2, The temperature hyperparameter $\sigma$ is set to 0.05, following [2]. For frame-level hyperparameters, an Adam optimizer is used to update MOF $\varepsilon_i$, using a learning rate $\beta$ of 8e-4. Following Frozen [2], we use batch sizes of 16 when fine-tuning on all downstream datasets, i.e., cross-modal video retrieval datasets.

D. Baselines

To justify the effectiveness of our MOF, we compare it with the following fourteen methods:

- **JSFusion [70]**: This is a cross-modal fusion method, which proposes a Joint Sequence Fusion model to fuse the video and text representation into a 3D tensor, and a convolutional network is employed to directly measure the similarity score.
- **Multi.Cues [34]**: This classic method is also designed for video retrieval tasks, which utilizes multi-modal features by a fusion strategy and a weighted triplet ranking loss to better learn the joint embedding.
- **ActBERT [71]**: This is a unified multi-modal transformer framework, which encodes global actions, local regional objects, and text sentences in a Transformer to improve language-and-visual alignment.
- **CE [35]**: This is a multimodal fusion method, which uses collaborative gating to fuse rich multimodal features to obtain a better video representation.
- **MMT [36]**: This is also a multimodal fusion approach, which uses BERT for text representation and proposes a multi-modal transformer to jointly encode diverse modalities in videos for video representation.
- **AVNet [39]**: This is a multimodal retrieval approach, which uses raw audio and other modalities to better align and perform video-text retrieval.
- **FSE [72]**: This is a cross-modal alignment approach, which proposes hierarchical modeling of videos and paragraphs.
- **T2VLAD [37]**: This is also a cross-modal alignment approach that introduces a paradigm of global-local alignment based on NetVLAD [73] to perform video retrieval.
- **SUPPORT-SET [40]**: This is a contrastive learning approach, which designs a generative objective to improve the instance discrimination limitations of contrastive learning.
- **TACo [74]**: This is also a contrastive learning approach that introduces token-aware cascade contrastive learning to improve the video-text alignment.
- **ClipBERT [1]**: This is an end-to-end retrieval approach, which leverages a sparse sampling strategy to train the model in an end-to-end manner.
- **CLIP4Clip-meanP [25]**: This is also an end-to-end retrieval approach, which uses CLIP [45] to extract the frame features and the text features, and then uses the mean pooling to aggregate the feature of all frames for video representation.
- **Frozen [2]**: This is an end-to-end pre-training approach, which uses the recent TimeSformer [16] as the visual encoder and designs a curriculum learning schedule to train the model on both image and video datasets.
- **TS2 [75]**: This is also an end-to-end retrieval approach, which designs token shift and selection transformer to dynamically adjust the token sequence and selects informative tokens in both temporal and spatial dimensions.

E. Experimental Results and Analyses

1) Comparisons With the State-of-The-Art Methods: We compare the text-to-video retrieval results obtained by different methods in Tables I, II, and III. The results show that utilizing our method as a plug-in module with the state-of-the-art [2] approach consistently yields the best performance on three datasets. To ensure a fair comparison, we reimplement the recently proposed Frozen [2] method with our downloaded MSVD and DiDeMo datasets. Notably, the improvements achieved relative to Frozen [2] can reach 2.2%, 2.1%, 1.9% on MSRVTT, MSVD, and DiDeMo, respectively, in terms of the R@10 metric. To further compare our approach with a recent CLIP-based

---

$^1$[Online]. Available: http://www.pytorch.org
work [25], we take our MOF method as a plug-in architecture for the CLIP4Clip method [25]. From Tables I, II, and III, we can see that CLIP4Clip w/ ours achieves comparable results with fewer frames and shorter test times. Furthermore, we observe that raw video-based end-to-end methods [2], [25] achieve better performance than video feature-based nonend-to-end methods [36], [40], demonstrating the advantage of pretrained visual transformers in video representation cases.

2) Ablation Study: We conduct an extensive ablation study on the number of sampled frames and different video backbones, as shown in Tables IV and VI.

Number of sampled frames: As shown in Table IV, we compare the performance of Frozen w/ ours by varying the number of frames on the MSR-VTT, MSVD, and DiDeMo datasets. In Table IV, we observe that the boost provided by our MOF training process increases under larger frame number settings; e.g., on MSR-VTT, Frozen w/ ours achieves a 71.7% R@10 (SF = 8 ⇒ 8) and a 72.8% R@10 (SF = 32 ⇒ 8). This result reveals that our MOF training procedure yields better performance by compressing more frames. On MSR-VTT, relative to settings with fewer frames, Frozen w/ ours (SF = 8 ⇒ 4) achieves comparable performance (70.2) to that of Frozen (70.6) under the same parameter settings while using fewer frames (4 vs 8). Additionally, the performance of the proposed approach further improves when compressing more frames (70.2 ⇒ 71.1). Another observation is that Frozen w/ ours (SF = 8 ⇒ 4) requires slightly more training time (9 vs 8) and less testing time (24 vs 46) than Frozen (SF = 8). Thus, Frozen w/ ours can provide a good tradeoff between effectiveness (i.e., retrieval accuracy) and efficiency (i.e., computing speed), with a significant impact on real-world problems such as cross-modal online video searches. Finally, we compare Frozen w/ ours (SF = 1 ⇒ 1) with Frozen (SF = 8) on MSR-VTT. We report that this variant of our model produces results that are approximately 13.1% worse than those of Frozen. We conjecture that the performance degradation observed with 1 frame is due to a reduction in temporal information. Due to GPU memory constraints, we are not able to train any models with R values higher than 32, as those models have much higher computational costs. Analogously, our method achieves new state-of-the-art performance on the MSVD and DiDeMo datasets.

Different video backbones: The core of our proposed MOF method is a novel meta-learning-based training paradigm that can be adapted to different video backbones. Therefore, to explore the robustness of our proposed MOF approach to different...
We conduct corresponding analytical experiments. On MSR-VTT, we use three types of video backbones, including ResNet-50, ResNet-152, and TimeSformer. The ablation results obtained with the different video backbones is shown in Table VI. To conduct fair comparisons, all models are not pretrained on WebVid-2M and are only finetuned on the MSR-VTT training set. From the results, we find that taking our MOF method as a plug-in module for TimeSformer yield the best performance. We note that ResNet-50 and ResNet-151 also achieve better results when used together with our MOF method, which indicates that our MOF module can be plugged into different video backbones.

Different baselines: As shown in Table V, we compare the experimental results obtained by plugging our MOF method into other baselines [25, 75]. Based on the results, we observe a similar trend to that of Frozen [2], where CLIP4Clip-meanP [25] and TS2 [75] achieve better performance by incorporating our MOF method as a plug-in module.
HAN et al.: EFFICIENT CROSS-MODAL VIDEO RETRIEVAL WITH META-OPTIMIZED FRAMES

Fig. 5. Qualitative examples of text-to-video retrieval. In (left), (right), we show the Top-5 ranks of Frozen [2] and our Frozen w/ ours on the MSR-VTT and MSVD datasets. Given a textual description as a query, we retrieve the most relevant video ranked from top to bottom. Ground-truth video is bounded in green box.

| Video backbone | Vis Enc.Init. | SF | R@1 | R@5 | R@10 |
|----------------|---------------|----|-----|-----|------|
| ResNet-50      | ImageNet-1k   | 2  | 2.0 | 5.0 | 10.4 |
| ResNet-50 w/ ours | ImageNet-1k   | 8  | 2.4 | 7.3 | 13.6 |
| ResNet-152     | ImageNet-1k   | 2  | 2.3 | 5.9 | 10.5 |
| ResNet-152 w/ ours | ImageNet-1k   | 8  | 3.3 | 10.4 | 15.0 |
| TimeSformer    | ImageNet-21k  | 2  | 14.0 | 36.9 | 52.0 |
| TimeSformer w/ ours | ImageNet-21k  | 8  | 15.1 | 41.7 | 54.3 |

Note that for fair comparisons, all models were not pretrained on WebVid-2M and only finetuned on MSR-VTT train set.

Fig. 6. Qualitative examples of text-to-video retrieval. In (left), (right), we show the Top-5 ranks of Frozen [2] and our Frozen w/ ours on the DiDeMo dataset. Given a textual description as a query, we retrieve the most relevant video ranked from top to bottom. Ground-truth video is bounded in green box.

| Parameterized frames | Video backbone | SF | R@1 | R@5 | R@10 |
|----------------------|----------------|----|-----|-----|------|
| video feature        | TimeSformer    | 2  | 10.3 | 32.5 | 44.1 |
| video feature w/ ours | TimeSformer    | 8  | 10.9 | 33.8 | 45.6 |
| raw video            | TimeSformer    | 2  | 14.0 | 36.9 | 52.0 |
| raw video w/ ours    | TimeSformer    | 8  | 15.1 | 41.7 | 54.3 |

Note that for fair comparisons, all models were not pretrained on WebVid-2M and only finetuned on MSR-VTT train set.

**Raw video-based frameworks versus video feature-based frameworks:** We also compare different parameterized frameworks in our MOF method, including a video feature-based framework and a raw video-based framework. As shown in Table VII, we obtain the following observations. 1) The performance drop exhibited by the “video feature” approach proves that the raw video-based framework contributes to cross-modal understanding and assists with improving the resulting retrieval accuracy. One possible reason for this drop is that these features are limited by the pre-extracted single-modal features and are not properly learned for the target downstream tasks. 2) The performance improvement provided by the “raw video” approach over the “video feature” approach demonstrates the positive impact of exploiting raw video-based representative “frames” in video retrieval tasks. However, considering effectiveness (i.e., retrieval accuracy), we ultimately settle on the raw video-based framework.

3) **Qualitative Analysis:** In Fig. 5(a), we visualize two qualitative examples of the text-to-video retrieval results obtained by Frozen and Frozen w/ ours on MSR-VTT. The text query describes multiple objects (e.g., “man”, “garlic”, and “pan”) and the action (“sprinkling”) in a short-term segment. In Fig. 5, the two videos ranked at the top contain similar sets of objects and actions. By using MOF, Frozen w/ ours successfully ranks the ground-truth video at the top. In Fig. 5(b) (left) shows an example where Frozen with 8 frames cannot distinguish videos with similar objects, and the action attributes are incorrectly predicted. The performance of Frozen w/ ours indicates that MOF learning enhances the expressiveness of the resulting video representation and further improves the achieved retrieval performance.

In Figs. 5(b) and 6, we provide additional qualitative text-to-video retrieval results obtained on MSVD [66] and DiDeMo [67]. Given a text query, in most cases, Frozen w/ ours successfully ranks the true positive in the first position.
objects and motions are incorrectly predicted. This is mainly because sufficient visual information in the sampled frames is missed. On DiDeMo, Fig. 6 also shows that the obtained results are in line with the conclusions determined on MSR-VTT and MSVD. Hence, the proposed MOF approach is effective in cross-modal video retrieval tasks.

4) Visualization Results: To intuitively observe the effectiveness of introducing the MOF training process, we visualize the attention map between the video frames and their corresponding sentences. We visualize the attention map from the output of the first transformer layer. Specifically, we use a text token as the query and visualize the attention weights provided for all spatial tokens. To analyze whether MOF can enhance the expressiveness of the video representation, in Fig. 7(a) and (b), we visualize the regular frames and the MOFs, where the regular frames are derived from different scenarios. We can see that Frozen w/ ours highlights key visual cues related to the text, e.g., the human hand, sandwich, and cooking tools, and ignores (the useless) background pixels such as home decorations. Furthermore, the MOFs have significantly more salient regions (orange) than each regular frame. This visualization verifies that the learned MOFs provide more useful visual semantic information, thus boosting the expressiveness of the video representation. This is the essence of our video compression idea.

5) Parameter Sensitivity: As shown in Fig. 8, we conduct parameter sensitivity experiments on our model. Specifically, we evaluate the impacts of the batch size parameter and the learning rate $\beta$ in (11). Notably, we omit the text-video retrieval results obtained on the other two datasets due to space limitations, but they exhibit show similar trends to those obtained on MSR-VTT [4]. To analyze the influence of the hyperparameter, that is, the batch size used for the MSR-VTT dataset, we train Frozen [2] with 2 frames and Frozen w/ ours (i.e., using our MOF training approach as a plug-in module) with 16 frames compressed into 2 frames. Fig. 8 (left) presents the results obtained with different batch sizes on the MSR-VTT dataset in terms of Recall@5; note that Recall@1 and Recall@10 present the same trend. We observe that increasing the batch size leads to consistent performance gains. Due to GPU memory constraints, we are not able to test our model on batch sizes larger than 64. Note that the curves of each parameter are obtained by fixing the other remaining parameters. To explore the effect of the learning rate $\beta$, we train Frozen w/ ours with a batch size of 16 to compress 16 frames into 2 frames on the MSR-VTT dataset. Fig. 8 (right) demonstrates that the best retrieval performance is achieved when $\beta = 0.0008$. Moreover, we also observe that the proposed method is more sensitive to $\beta$ than the competing approach, which demonstrates the importance of the learning rate $\beta$ for obtaining meta-optimized frames.

VI. CONCLUSION

In this paper, we proposed a novel meta-learning method Meta-Optimized Frames (MOF) that automatically compresses the video into fewer but more representative frames. In essence, our learned “meta-optimized frames” are optimizable and adaptable, and can improve the training flexibility of online retrieval systems. We conducted extensive experiments on three video
retrieval benchmarks and validated that our MOF achieves competitive multi-modal retrieval performance while greatly reducing the inference costs. In addition, our training method of MOF is generic and could be potentially incorporated into other video tasks, e.g., video captioning and video classification, to improve the model performance. We will take these as our future works.

REFERENCES

[1] J. Lei et al., “Less is more: Clipbert for video-and-language learning via sparse sampling,” in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit., 2021, pp. 7331–7341.

[2] M. Duan, A. Nagrani, G. Vural, and A. Zisserman, “Frozen in time: A joint video and image encoder for end-to-end retrieval,” in Proc. IEEE Int. Conf. Comput. Vis., 2021, pp. 1728–1738.

[3] A. J. Wang et al., “Object-aware video-language pre-training for retrieval,” in Proc. CVPR, 2022, pp. 3313–3322.

[4] J. Xu, T. Mei, T. Yao, and Y. Rui, “Msr-vtt: A large video description dataset for bridging video and language,” in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit., 2016, pp. 5288–5296.

[5] C. Feichtenhofer, H. Fan, J. Malik, and K. He, “Slowfast networks for video recognition,” in Proc. IEEE Int. Conf. Comput. Vis., 2019, pp. 6202–6211.

[6] W. Lu, D. Li, L. Nie, P. Jing, and Y. Su, “Learning dual low-rank representation for multi-label micro-video classification,” IEEE Trans. Multimedia, vol. 25, pp. 77–89, 2023.

[7] H. Zhang, W. Mei, N. Nie, and S. Jiang, “Hybrid-attention enhanced two-stream fusion network for video venue prediction,” IEEE Trans. Multimedia, vol. 23, pp. 2917–2929, 2021.

[8] H. Zhang, Y. Hao, and C.-W. Ngo, “Token shift transformer for video classification,” in Proc. 29th ACM Int. Conf. Multimedia, 2021, pp. 917–925.

[9] F. Chen, J. Shao, Y. Zhang, X. Xu, and H. T. Shen, “Interclass-relativity-adaptive metric learning for cross-modal matching and beyond,” IEEE Trans. Multimedia, vol. 23, pp. 3073–3084, 2021.

[10] X. Li, F. Zhou, C. Xu, J. Ji, and G. Yang, “Sea: Sentence encoder assembly for video retrieval by textual queries,” IEEE Trans. Multimedia, vol. 23, pp. 4351–4362, 2021.

[11] M. Qi, J. Qin, Y. Yang, and J. Luo, “Semantics-aware spatial-temporal binaries for cross-modal video retrieval,” IEEE Trans. Image Process., vol. 30, pp. 2989–3004, 2021.

[12] Y. Wang et al., “Siameq alignment network for weakly supervised video moment retrieval,” IEEE Trans. Multimedia, vol. 25, pp. 3921–3933, 2023.

[13] X. Yang, F. Peng, W. Ji, M. Wang, and T.-S. Chua, “Deconfounded video moment retrieval with causal intervention,” in Proc. 44th Int. ACM SIGIR Conf. Res. Develop. Inf. Retrieval, 2021, pp. 1–10.

[14] X. Yang et al., “Video moment retrieval with cross-modal neural architecture search,” IEEE Trans. Image Process., vol. 31, pp. 1204–1216, 2022.

[15] A. Dosovitskiy et al., “An image is worth 16x16 words: Transformers for image recognition at scale,” in Proc. Int. Conf. Learn. Representations, 2021.

[16] A. J. Wang et al., “Object-aware video-language pre-training for retrieval,” in Proc. Int. Conf. Learn. Representations, 2021.

[17] J. Wang, X. Yang, H. Li, Z. Wu, and Y.-G. Jiang, “Efficient video-text retrieval via image clip,” in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit., 2020, pp. 9876–9886.

[18] A. Rouditchenko et al., “Avlnet: Learning audio-visual language representations from instructional videos,” 2020, arXiv:2006.09199.

[19] M. Patrick et al., “Support-set benchmarks for video-text representation learning,” in Proc. Int. Conf. Learn. Representations, 2021.

[20] H. Fang, P. Xiong, L. Xu, and Y. Chen, “Clip2video: Mastering video-text retrieval via image clip,” 2021, arXiv:2106.11007.

[21] X. Cheng, H. Lin, X. Wu, F. Yang, and D. Shen, “Improving video-text retrieval by multi-stream corpus alignment and dual softmax loss,” 2021, arXiv:2109.04290.

[22] Y. Ma et al., “X-clip: End-to-end multi-grained contrastive learning for video-text retrieval,” in Proc. 30th ACM Int. Conf. Multimedia, 2022, pp. 638–647.

[23] S. K. Gorti et al., “X-pool: Cross-modal language-video attention for text-video retrieval,” in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit., 2022, pp. 5006–5015.

[24] A. Radford et al., “Learning transferable visual models from natural language supervision,” in Proc. Int. Conf. Mach. Learn., 2021, pp. 8748–8763.

[25] Q. Wang, Y. Zhang, Y. Zheng, P. Pan, and X.-S. Hua, “Disentangled representation learning for text-video retrieval,” 2022, arXiv:2203.07111.

[26] V. S. Heinrich and H. V. Stackelberg, The Theory of The Market Economy. Oxford, U.K.: Oxford Univ. Press, 1952.

[27] I. Goodfellow et al., “Generative adversarial nets,” NIPS, 2014.

[28] T. Wang, J.-Y. Zhu, A. Torralba, and A. A. Efros, “Dataset distillation,” 2018, arXiv:1811.10959.

[29] Y. Liu, B. Schiele, and Q. Sun, “Adaptive aggregation networks for class-incremental learning,” in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit., 2021, pp. 2544–2553.

[30] B. Zhu, C.-W. Ngo, J. Chen, and Y. Hao, “R2gan: Cross-modal recipe retrieval with generative adversarial network,” in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit., 2019, pp. 11477–11486.

[31] J. Wei, X. Xu, Z. Wang, and G. Wang, “Meta self-paced learning for cross-modal matching,” in Proc. 27th ACM Int. Conf. Multimedia, 2021, pp. 3835–3843.

[32] T. Hospedales, A. Antoniou, P. Micacelli, and A. Storkey, “Meta-learning in neural networks: A survey,” IEEE Trans. Pattern Anal. Mach. Intell., vol. 44, no. 9, pp. 5149–5169, Sep. 2021.
[54] A. Antoniou, H. Edwards, and A. Streetey, “How to train your maml,” in Proc. ICLR, 2019, pp. 1–12.

[55] J. Lorraine, P. Vicol, and D. Duvenaud, “Optimizing millions of hyperparameters by implicit differentiation,” in Int. Conf. Artif. Intell. Statist., 2020, pp. 1540–1552.

[56] N. C. Mithun, J. B. Li, F. Metze, A. K. Roy-Chowdhury, and S. Das, “Cmu-ucr-bosch, trecvid 2017: Video to text retrieval,” in Proc. TRECVID, 2017.

[57] Z. Qiu, T. Yao, Y. Shu, C.-W. Ngo, and T. Mei, “Condensing a sequence to one informative frame for video recognition,” in Proc. IEEE Int. Conf. Comput. Vis., 2021, pp. 16311–16320.

[58] E. Elhamifar, G. Sapiro, and S. S. Sastry, “Dissimilarity-based sparse subset selection,” IEEE Trans. Pattern Anal. Mach. Intell., vol. 38, no. 11, pp. 2182–2197, Nov. 2015.

[59] H. Bilen, B. Fernando, E. Gavves, A. Vedaldi, and S. Gould, “Dynamic image networks for action recognition,” in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit., 2016, pp. 3034–3042.

[60] M. Tavakolian, M. Sabokrou, and A. Hadid, “Aved: Adversarial video distillation,” 2019, arXiv:1907.05640.

[61] D. Chen et al., “Cross-modal retrieval with heterogeneous graph embedding,” in Proc. 30th ACM Int. Conf. Multimedia, 2022, pp. 3291–3300.

[62] X. Lu, L. Zhu, Z. Cheng, L. Nie, and H. Zhang, “Online multi-modal hashing with dynamic query-adaption,” in Proc. 42nd Int. ACM SIGIR Conf. Res. Develop. Inf. Retrieval, 2019, pp. 715–724.

[63] X. Lu, L. Liu, L. Nie, X. Chang, and H. Zhang, “Semantic-driven interpretable deep multi-modal hashing for large-scale multimedia retrieval,” IEEE Trans. Multimedia, vol. 23, pp. 4541–4554, 2021.

[64] Y. Hu, M. Liu, X. Su, Z. Gao, and L. Nie, “Video moment localization via deep cross-modal hashing,” IEEE Trans. Image Process., vol. 30, pp. 4667–4677, 2021.

[65] A. Zhai and H.-Y. Wu, “Classification is a strong baseline for deep metric learning,” 2018, arXiv:1811.12649.

[66] D. Chen and W. B. Dolan, “Collecting highly parallel data for paraphrase evaluation,” in Proc. 49th Annu. Meeting Assoc. Comput. Linguistics, Hum. Lang. Technol., 2011, pp. 190–200.

[67] L. Anne Hendrickx et al., “Localizing moments in video with natural language,” in Proc. IEEE Int. Conf. Comput. Vis., 2017, pp. 5803–5812.

[68] S. Venugopalan et al., “Sequence to sequence-video to text,” in Proc. IEEE Int. Conf. Comput. Vis., 2015, pp. 4534–4542.

[69] P. Sharma, N. Ding, S. Goodman, and R. Soricut, “Conceptual captions: A cleaned hypernymed, image alt-text dataset for automatic image captioning,” in Proc. Assoc. Comput. Linguistics, 2018, pp. 2556–2565.

[70] Y. Yu, J. Kim, and G. Kim, “A joint sequence fusion model for video question answering and retrieval,” in Proc. Eur. Conf. Comput. Vis., 2018, pp. 471–487.

[71] L. Zhu and Y. Yang, “Actbert: Learning global-local video-text representations,” in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit., 2020, pp. 8746–8755.

[72] B. Zhang, H. Hu, and F. Sha, “Cross-modal and hierarchical modeling of video and text,” in Proc. Eur. Conf. Comput. Vis., 2018, pp. 374–390.

[73] R. Arandjelovic, P. Gronat, A. Torii, T. Pajdla, and J. Sivic, “Netvlad: Cnn architecture for weakly supervised place recognition,” in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit., 2016, pp. 5297–5307.

[74] J. Yang, Y. Bisk, and J. Gao, “Taco: Token-aware cascade contrastive learning for video-text alignment,” in Proc. IEEE Int. Conf. Comput. Vis., 2021, pp. 11562–11572.

[75] Y. Liu, P. Xiong, L. Xu, S. Cao, and Q. Jin, “Ts2-net: Token shift and selection transformer for text-video retrieval,” in Proc. ECCV, 2022, pp. 319–335.

Ning Han received the Ph.D. degree in computer science from Hunan University, Changsha, China, in 2023. He was a visiting Ph.D. student with Living Analytics Research Centre, Singapore Management University, Singapore. He is currently a Lecturer with the School of Computer Science, Xiangtan University, Xiangtan, China. He has authored or coauthored several research papers in top conferences, including SIGIR and ACM MM. His research interests include multimedia contents analysis including cross-modal retrieval, multimodal fusion, and video recognition.

Xun Yang received the Ph.D. degree from the Hefei University of Technology, Hefei, China, in 2017. From 2015 to 2017, he visited the University of Technology Sydney (UTS), Sydney, NSW, Australia, as a joint Ph.D. Student. He was a Research Fellow with NExT++ Research Center, National University of Singapore (NUS), Singapore, from 2018 to 2021. He is currently a tenure-track Professor with the Department of Electronic Engineering and Information Science, University of Science and Technology of China (USTC), Hefei. His research interests include information fusion, cross-media analysis and reasoning, and computer vision. He is a regular PC Member and Invited Reviewer for top-tier conferences and prestigious journals in multimedia and artificial intelligence, such as ACM Multimedia, IJCAI, AAAI, CVPR, and ICCV. He was a SPC Member for AAAI 2021.

Ee-Peng Lim received the Ph.D. degree from the University of Minnesota, Minneapolis, MN, USA. He is currently the Lee Kong Chian Professor with the School of Computing and Information Systems, Singapore Management University (SMU), Singapore. His research expertise covers social media mining, social/urban data analytics, and information retrieval. He has authored or coauthored more than 400 international journal and conference papers from his research works. He was the recipient of the Distinguished Contribution Award at the 2019 Pacific Asia Conference on Knowledge Discovery and Data Mining (PAKDD) and Test of Time Award at the 2020 ACM International Conference of Web Search and Data Mining (WSDM). He is on Singapore’s Social Science Research Council which focuses on developing talent and strengthening social science and humanities research that benefits social and economic development in Singapore and the Asian region.

Hao Chen received the Ph.D. degree from Central South University, Changsha, China, in 2012. He is currently a Professor and Ph.D. Supervisor with the College of Computer Science and Electronic Engineering, Hunan University, Changsha. His research interests include recommender systems, data mining, and multimedia information retrieval. His work has appeared in top venues. He was a reviewer for various journals and conferences.

Qianru Sun (Member, IEEE) received the Ph.D. degree from Peking University, Beijing, China, in 2016, advised by Prof. Hong Liu. From 2014 to 2015, she was a visiting Ph.D. student advised by Prof. Tat-suyu Harada with the University of Tokyo, Tokyo, Japan. She held the Lise Meitner Award Fellowship and was a Postdoctoral Researcher working with Prof. Dr. Bernt Schiele and Dr. Mario Fritz with the Max-Planck Institute for Informatics (MPII), Saarbrücken, Germany, for two years, where she currently has a visiting Researcher position. She was a Research Fellow with Prof. Tat-Seng Chua and leads the Computer Vision Group with the NExT++ Lab, National University of Singapore, Singapore, since 2018. She is an Assistant Professor with the School of Information Systems, Singapore Management University, Singapore, since 2019. Her research interests include computer vision and machine learning. Specific topics include image recognition, conditional image generation, meta-learning, and transfer learning.

Authorized licensed use limited to the terms of the applicable license agreement with IEEE. Restrictions apply.