A Large-scale Comprehensive Dataset and Copy-overlap Aware Evaluation Protocol for Segment-level Video Copy Detection

Sifeng He*, Xudong Yang*, Chen Jiang*, Gang Liang, Wei Zhang, Tan Pan, Qing Wang, Furong Xu,
Chunguang Li, Jingxiong Liu, Hui Xu, Kaiming Huang, Yuan Cheng, Feng Qian†, Xiaobo Zhang†, Lei Yang
Ant Group
{sifeng.hsf*, jiegang.yxd*, qichen.jc*, youzhi.qf†, ayou.zxb†}@antgroup.com

Abstract

In this paper, we introduce VCSL (Video Copy Segment Localization), a new comprehensive segment-level annotated video copy dataset. Compared with existing copy detection datasets restricted by either video-level annotation or small-scale, VCSL not only has two orders of magnitude more segment-level labelled data, with 160k realistic video copy pairs containing more than 280k localized copied segment pairs, but also covers a variety of video categories and a wide range of video duration. All the copied segments inside each collected video pair are manually extracted and accompanied by precisely annotated starting and ending timestamps. Alongside the dataset, we also propose a novel evaluation protocol that better measures the prediction accuracy of copy overlapping segments between a video pair and shows improved adaptability in different scenarios. By benchmarking several baseline and state-of-the-art segment-level video copy detection methods with the proposed dataset and evaluation metric, we provide a comprehensive analysis that uncovers the strengths and weaknesses of current approaches, hoping to open up promising directions for future works. The VCSL dataset, metric and benchmark codes are all publicly available at https://github.com/alipay/VCSL.

1. Introduction

In recent years, the wide spread of pirated multimedia has attracted attention from both users and platforms all over the world. The dramatic rise of pirated content has been fueled by the large amounts of user-generated content (UGC) and professionally-generated content (PGC) uploaded to content sharing market, e.g., over 500 hours of video are uploaded to YouTube every minute [1] , the average monthly paying users of Bilibili have increased by 62% in one year alone [2]. These videos can generate significant advertising revenue, providing strong incentive for unscrupulous individuals that wish to capitalize on this bonanza by skillful copyright infringement [3]. Some video editing specialists even design methods to evade infringement detection algorithms by cropping, melting, and merging short clips from popular videos which make accurate copy detection even more challenging. Given the escalating adversarial relation between the platform algorithm and evolved piracy, comprehensive datasets with real partial video infringement become increasingly essential.

Besides copyright protection, a video copy detection (VCD) system is important in applications like video classification, tracking, filtering and recommendation [4–6]. In most cases, video-level copy detection results alone are not sufficient as the detected videos are usually displayed and interacted with system users for downstream tasks. Hence, designing an approach that can locate the copied segments is preferred and has already attracted lots of attentions in recent works [7–11].

However, manually annotating copied segments between videos is time-consuming and costly. Some datasets for copy detection, e.g., CCWEB [12], FIVR [13] and SVD [14] provide only video-level annotation indicating whether two videos contain copied parts or not, which is coarse-grained and incapable of evaluating segment-level copy detection methods. Other datasets, e.g., MUSCLE-VCD [15] and TRECVID [16], produce automatically segment-level labels by generating simulated copied segments with pre-defined transformations, which may not be representative for real-world data [11]. The only manually-labelled segment-level dataset, VCDB [11] released in 2014, contains only 6k labelled videos pairs with 9k segment pairs, and over 70% of copy durations are less than 1minute. The annotation quantity and video diversity of existing datasets are not sufficient to develop segment-level video copy detection algorithms that need training data and labels.

To address these issues, we present a comprehensive dataset, VCSL, specifically designed for segment-level

*These authors contributed equally to this research.
†Corresponding author.

video copy detection. This dataset, which will be made publicly available, contains over 160k infringed video pairs with 280k carefully annotated segment pairs. All of these videos are realistic copies from Youtube or Bilibili, which cover a wide range of video topics including movies, music videos, sports, etc.

Meanwhile, existing evaluation protocols for segment-level video copy detection exhibit an obvious drawback that most of them utilize ground-truth copied segments as queries rather than the entire videos [7, 8, 11]. This is unpractical for real copy detection scenario where it is hard to know a priori that which part of a video will be pirated. Hence, we indicate a protocol that is more realistic with a pair of copied videos as input, and a new metric is jointly proposed to address previous unreasonable issues. With the awareness of both copied segments and overlap accuracy, our proposed metric fully considers the distinctiveness regarding the segment division equivalence (illustrated in Sec.4.1) of copy detection task, and it is more suitable and robust for various infringe situations.

Furthermore, we introduce a benchmark for segment-level video copy detection. We decouple the entire algorithm process into two main algorithm modules: feature extraction and temporal alignment. Then we evaluate the baseline and state-of-the-art (SOTA) algorithms of both parts on the splitted test set of VCSL. The components described above represent a complete benchmark suite, providing researchers with the necessary tools to facilitate the evaluation of their methods and advance the field of segment-level video copy detection.

2. Related Work

In this section we provide an overview of datasets and evaluation metrics designed for different video copy detection and retrieval tasks, followed by a survey of techniques targeting segment-level video copy detection.

2.1. Datasets and Evaluation

We briefly review the datasets for VCD task in this section including CCWEB [12], MUSCLE VCD [15], TRECVID [16], FIVR [13], SVD [14] and VCDB [11].

CCWEB [12] dataset is one of the most widely used dataset. It contains 24 query videos and 12,790 labelled videos. All retrieved videos in the video sets were manually annotated by three annotators based on their video-level relation to the query video. In addition to only video-level annotation, CCWEB also shows limitations on both video transformation and topic category diversity, and almost all the recent methods can achieve a near-perfect performance (video-level mAP of $>$0.99) on this dataset.

MUSCLE-VCD [15] collects 18 videos to construct query set. Then the authors utilize query videos to generate 101 videos as labeled set based on some predefined transformations. Similarly, TRECVID datasets [16] were constructed following the same process as the MUSCLE-VCD dataset. The latest edition of the dataset contains 11,503 reference videos of over 420 hours and 11,256 queries. The queries were automatically generated by randomly extracting a segment from a dataset video and imposing a few predefined transformations. Therefore, copies in MUSCLE-VCD and TRECVID are all simulated based on predictable processing, which are less diversified and much easier to be detected and retrieved.

FIVR [13] consists of 225,960 videos and 100 queries. This dataset collects fine-grained incident retrieved videos including three retrieval tasks: a) the Duplicate Scene Video Retrieval (DSVR), b) the Complementary Scene Video Retrieval (CSVR), and c) the Incident Scene Video Retrieval (ISVR). Despite the large-scale video collection of FIVR, only part of the first task (about 1325 annotated videos) in this dataset is relevant to the scope of video infringement, and all the videos are news event labelled in video-level. Similarly, SVD [14] is also a large-scale near duplicated dataset with only video-level annotation, and most of videos in SVD are less than 20 seconds.

The most relevant dataset to our work is VCDB dataset [11] consisting of 28 query sets and 528 labelled videos with 9,236 pairs of copied segments. The annotation gives the precise temporal location of each copied pair and thus is suitable for segment-level copy detection task. However, a couple of obvious weaknesses exist in VCDB. First, the amount of both labelled video data and positive pairs are too limited for further usage. Some topic categories contain only one query set, and it is impossible to split train and test set from it. Over 90% of the videos in VCDB is less than 3 minutes which is also lack of diversity on duration.

Table 1 summarizes some statistics of the aforementioned datasets. Currently, there is no dataset that simultaneously supports segment-level annotation, real copies collection and large-scale diversity. This motivates us to build a comprehensive video partial copy dataset.

In the aspect of evaluation metric, the video-level evaluation metric (mAP) of VCD task has been well discussed in previous work [5, 12] and will not be covered in this paper. Previous segment-level evaluation metrics are introduced with MUSCLE-VCD [15] and VCDB datasets [11]. Most of recent research works [7–9] adopt segment precision and recall defined in VCDB as follows:

$$\begin{aligned} SP &= \frac{|correctly\ detected\ segments|}{|all\ detected\ segments|} \\ SR &= \frac{|correctly\ detected\ segments|}{|groundtruth\ copy\ segments|} \end{aligned} \tag{1}$$

In addition, VCDB also introduces a frame-level metric as auxiliary criteria:

Table 1. Comparison between VCSL and existing datasets. As we cannot access MUSCLE-VCD and TRECVID datasets, some statistics of these two datasets are N/A. The segment statistics (last two rows) of datasets with only video-level annotation (CCWEB, FIVR, SVD) are also N/A. ([1]) denotes the total of near-duplicated, duplicated scene, complementary scene and incident scene labels in FIVR. ([2]) indicates only near-duplicated video pair as copied videos. All the durations are calculated on labelled videos.

Item	CCWEB	MUSCLE-VCD	TRECVID	FIVR	SVD	FIVR-PVCD	VCDB	**VCSL**
Segment-level annotation	✗	✓	✓	✗	✗	✓	✓	✓
Type of copies	Realistic	Simulated	Simulated	Realistic	Realistic	Realistic	Realistic	Realistic
#query sets	24	18	11256	100	1206	100	28	122
#labelled videos	12,790	101	11,503	12,868[1]	34,020	5,964	528	9,207
Average duration (in second)	151.02	3564.36	131.44	113.12	17.33	113.12	72.77	364.90
#positive video pairs	3,481	N/A	N/A	1,325[2]	5935	10,211	6,139	**167,508**
#copied segments	N/A	N/A	N/A	N/A	N/A	10,870	9,236	**281,182**
Total copied duration (in hours)	N/A	N/A	N/A	N/A	N/A	76.4	326.8	**17,416.2**

$$FP = \frac{|correctly\ detected\ frames|}{|all\ detected\ frames|}$$
$$FR = \frac{|correctly\ detected\ frames|}{|groundtruth\ copy\ frames|} \quad (2)$$

However, both segment P/R and frame P/R have their limitations. The most significant one is that the protocol uses each segment in copy pairs rather than entire video as a query. Meanwhile, for segment P/R metric, a detected segment pair is considered correct as long as both of them have at least one frame overlap with the ground truth pair, leading to a poor awareness of copy overlap and alignment accuracy. Hence, we need to unify the detection performance and alignment accuracy into one metric, and make it appropriate for different infringement scenarios.

2.2. Methods

Frame-level features are proved to gain a large margin in video retrieval tasks [5, 17] and are necessary to precisely locate copied segments. Current methods employ Deep CNNs [18] and Deep Metric Learning (DML) [17, 19] to extract robust features. The application of Maximum Activation of Convolutions (MAC) and its variants [18, 20] has proved to be an efficient representation in retrieval tasks. Recently, transformer [21] has recently emerged as an alternative to CNNs for visual recognition task [22,23]. Self-supervised pretrained transformer model shows competitive performance on image copy detection task [23].

After obtaining frame-level feature representation, a temporal alignment module needs to reveal the similarity and time range of one or multiple copied segments between the potential copied video pair. A simple method is to vote temporally by Temporal Hough Voting [11, 24]. The graph-based Temporal Network (TN) [25] takes matched frames as nodes and similarities between frames as weights of links to construct a network, and the matched clip is the weighted longest path in the network. Another method is Dynamic programming [26] to find a diagonal blocks with the largest similarity. Inspired by temporal matching kernel [27], LAMV [7] transforms the kernel into a differentiable layer to find temporal alignments. SPD [8] formulates temporal alignment as an object detection task on the frame-to-frame similarity matrix, achieving a state-of-the-art segment-level copy detection performance.

3. Dataset

3.1. Annotation

The dataset is constructed for meeting the following requirements: 1) the video copy transformations should be as diverse as possible, but avoid excessive transformations leading to extremely low image quality. 2) the category should cover most of common video topics; 3) the duration of video should not be limited to only one type (short video or long video). Based on the above requirements, we start from 122 carefully selected seed videos containing both PGC and UGC from Youtube and Bilibili. Each seed video is associated with a text query (keywords), which will be used to search potential relevant videos online. These 122 queries contain 11 common topics, i.e., movies, TV series, music videos, sports, games, variety show, animation, daily life, advertisement, news and kichiku. According to these seed videos and their corresponding text queries, around 100 videos per query are collected as potential copied videos from Youtube and Bilibili platform.

Different from previous video-level annotation, extracting and annotating copied segments from their parent videos is an extremely error-prone and sophisticated task, especially for some short copied clips in currently fashionable ”kichiku” videos. Therefore, we design a coordination process between algorithm engineers (us) and annotators with multiple steps shown in Fig.1. In terms of annotation cost, we employed 30 full-time well-trained annotators and spent about 4 months to finish entire annotation process (about 20,000 man-hours). All the following annotation steps contain a round of labelling by one annotator,

a round of quality check by another annotator and a final round of spot check by us.

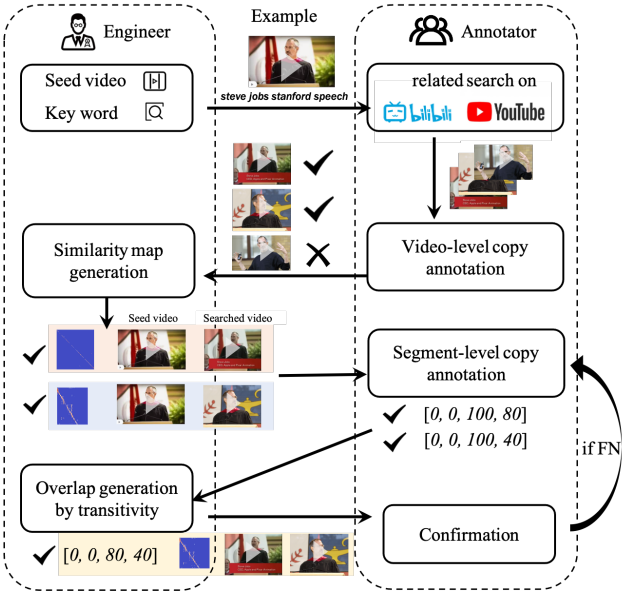


Figure 1. Overview of annotation process. Engineers (authors of this paper) in the left provide the initial query sets, build the annotation tool and clean the annotated data for next-step annotation. Annotators finish related search online, video-level and segment-level annotation. The similarity map is generated by dot production of frame-level features between the video pair. The segment annotation is expressed as a list with [*start timestamp of seed video, start timestamp of searched video, end timestamp of seed video, end timestamp of searched video*].

As mentioned before, the first step of annotation is to provide the seed videos and text queries to all annotators, then require them to search the related videos and give us a coarse video-level copy result. The second step is segment-level annotation on the filtered copied videos in each query set from the first step. At this step, precisely locating the matched segment temporal boundaries is extremely time-consuming, and even the experienced annotator can only finish 2-3 video pairs within one hour. Here, we build an annotation tool, which shows not only the original video pairs but also the frame-to-frame similarity map (illustrated in Sec.S1 of Supplementary Material in detail) as auxiliary assist to annotators. By observing approximate straight lines in the similarity map [8], the annotators can easily check missing copied clips after video comparison. After segment-level annotation, we obtain the segment copy information between one seed video and all the searched videos for each query set. Similar to VCDB datasets [11], we utilize transitivity property of the video copies to automatically generate new copied segments between the searched videos if their matched segments in the seed video intersect. It is notable that annotating copied segments between two videos relevant to the same seed video with the transitivity property can bring both false positives (e.g. the copied segments share no common contents) and false negative (e.g. there exist copied segments not appeared in the seed video) annotations. Therefore, in the final step, the annotators first check if the copied segments from transitivity propagation are correct matches and refine the copy boundaries. Then they are provided with the two videos and the similarity map similar to that in the second step to find potential copied segments out of the remaining area. All the above annotation process is shown in Fig.1.

3.2. Statistics

In total, we collect 9207 copied videos associated with 122 selected video queries from Bilibili and Youtube. After several rounds of carefully cooperated annotation by us and annotators, we extract and label 167,508 copied video pairs and 281,182 copied segments, both of which are two orders of magnitude more than the only realistic segment-level dataset VCDB. The detailed comparison between different datasets is shown in Table.1. It can be observed from the last row of Table.1 that the total copied duration of all segments in VCSL is even larger than total video duration in most of public available datasets, which shows the considerable large-scale of our dataset.

Fig.2 further shows some statistics of VCSL, and some detailed comparisons with VCDB. Different from VCDB and other short video datasets, VCSL contains videos longer than 30min, and these long videos include TV series and movies that are easy to be infringed nowadays. Meanwhile, the copied segment duration also covers a wider range from less than 5 seconds to even larger than 30min. Among the video pairs that have at least one segment copy, as high as 30% of them contain two or more copied segments and 45% of these segments are shorter than 1/5 of their parent video. All 122 video query sets are divided into 11 topics and the smallest topic contains at least three query sets for satisfying train-val-test splits. The least number of video copy pairs shown in Fig.2(e) for each topic is over 4k, which is more than half of labelled data (6k) from VCDB. Moreover, VCSL covers lots of realistic spatial and temporal transformations, and we list some in the Sec.S2 of Supplementary Material. The breadth and diversity of VCSL enable thorough comparisons between segment-level VCD approaches and make it possible to train supervised learning methods for which training data and labels are necessary.

4. Evaluation Protocol

4.1. Background and Motivation

As mentioned before, previous protocols use only ground-truth copied segments as queries which is inappropriate for practical use. Instead, we refine the evaluation

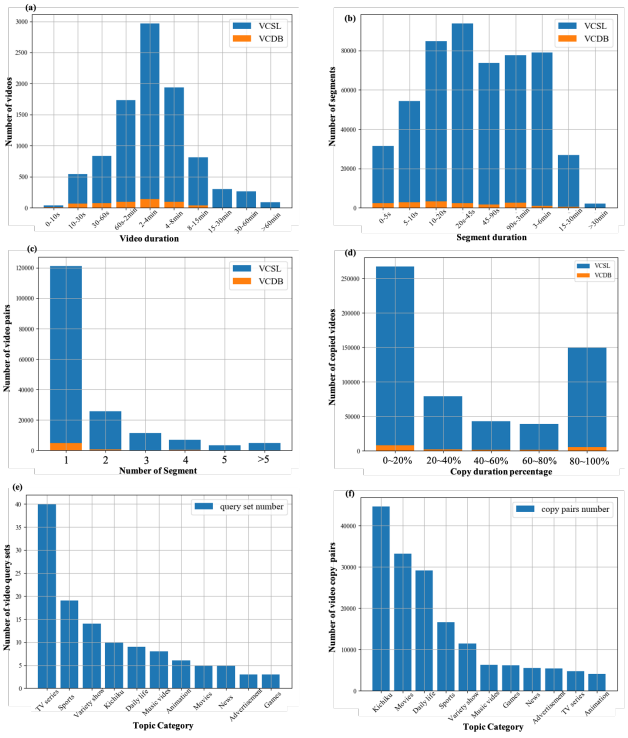


Figure 2. Data distribution of VCSL. All the blue bars represent the amount of VCSL and orange are for VCDB. (a) the numbers of videos in different video duration; (b) the numbers of segments in different segment duration; (c) the number of segments per video pair; (d) the duration percentage of the copy segments in their corresponding parent videos; (e) the number of video query sets for each topic category; (f) the number of video copy pairs for each topic category.

protocol by taking two entire videos as input and the system need to detect all the potential copied segments between the two videos. In this setting, most of previous metrics are inapplicable or need to be extended. As a result, we design a new metric to address this.

However, the evolved evaluation protocol brings new difficulties for designing the metric. During the annotation process, we observe that the boundaries of copied segments are hard to determine in some cases. As an example shown in Fig.3(a), some intermediate frames are edited or briefly inserted by other video frames, leading to ambiguous segment boundaries. Other common cases are mashup videos shown in Fig.3(b). If one single entire copied segment pair and a sequence of consecutive sub-segment pairs occupy the same copied part on original video pairs, we believe that these two annotations are both reasonable and correct. This also applies to predictions of algorithms with different inductive biases. The equivalence of an entire copied segment pair and its division of consecutive sub-segment pairs, i.e., segment division equivalence, must be taken into account when designing the new metric.

We decide to use the precision and recall as the evaluation metric because they are widely adopted. But calculating the recall and precision similar to previous metrics in Eq.(1) and Eq.(2) is problematic. The segment-level precision and recall in Eq.(1) fail to measure the overlap of copied segments, and the frame-level precision and recall in Eq.(2) calculated respectively on two videos give totally wrong results in some cases which are demonstrated in Fig.4(f). A better way to evaluate the recall and precision in this scenario is explained in the next section.

4.2. New Metric

The calculation of the metric can be more clearly described on the frame-to-frame similarity map between a video pair shown in Fig.4. For ease of representation, all the copied segment correspondences are depicted as bounding boxes in Fig.4 (a-f), and the copied pattern in similarity map in Fig.4 (c-f) are shown in oblique straight lines which represent the temporal sequential copy between two videos. The predicted and ground truth segment pairs inside the video pair are respectively denoted as predicted bounding boxes $\{P_j\}_{1,2,...,n}$ and GT bounding boxes $\{G_i\}_{1,2,...,m}$ shown in Fig.4.

Specifically, we firstly define the overlapped regions between all the predicted bounding boxes $\{P_j\}_{1,2,...,n}$ and each GT bounding box G_i with Intersect-over-Union (IoU) > 0 as:

$$O_i = \{P_1 \cap G_i, P_2 \cap G_i, ..., P_n \cap G_i\} \tag{3}$$

Then for each GT box G_i, we calculate the union length $L^x_{O_i}$ and $L^y_{O_i}$ of projected lines (which indicate frames) from O_i along both x/y axis (which indicate temporal axis of video A and video B). This process can be shown in Fig.4(a). We can also obtain the width and height of G_i as $L^x_{G_i}$ and $L^y_{G_i}$.

Therefore, the recall metric of a video pair is defined as follows:

$$Recall = \frac{\sum_{i=0}^{m} L^x_{O_i}}{\sum_{i=0}^{m} L^x_{G_i}} \cdot \frac{\sum_{i=0}^{m} L^y_{O_i}}{\sum_{i=0}^{m} L^y_{G_i}} \tag{4}$$

It is notable that we utilize the projection length on x and y axis rather than the bounding box area that is more commonly used in IoU [28,29]. This is to make the metric more robust against the equivalence of a single bounding box and its division of temporally consecutive bounding boxes mentioned above, which will be discussed later in Fig.4(d-e).

Similarly, for the precision metric, we first calculate the overlapped regions between all the GT bounding boxes $\{G_i\}_{1,2,...,m}$ and each predicted bounding box P_j,

$$O_j = \{G_1 \cap P_j, G_2 \cap P_j, ..., G_n \cap P_j\} \tag{5}$$

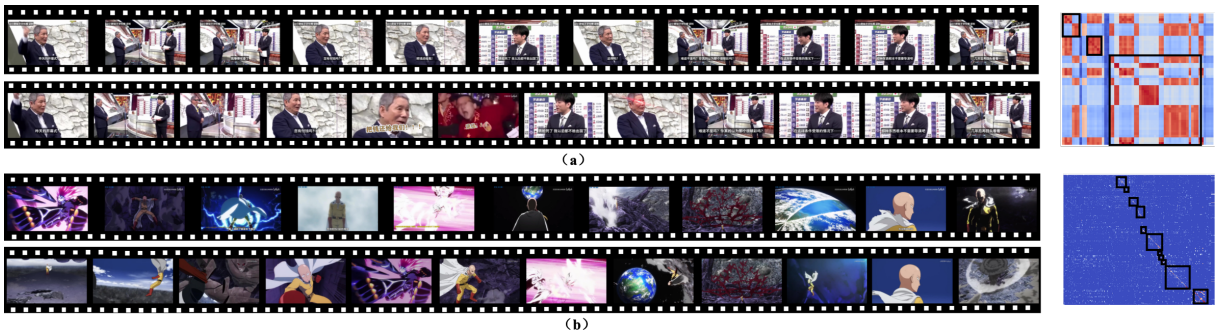
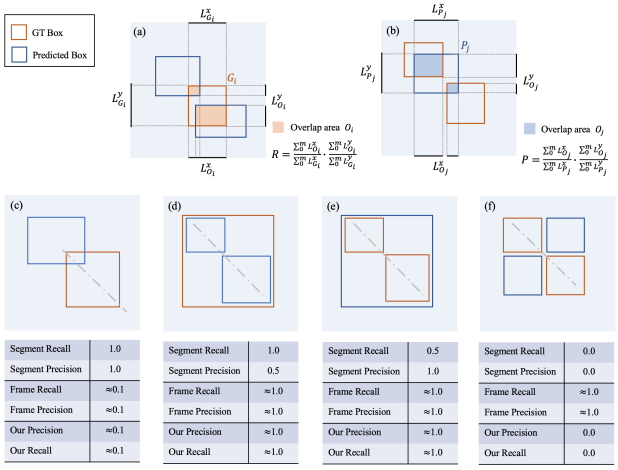


Figure 3. Two copied video examples with ambiguous segment boundaries. The left part is the timing screenshots of both videos, and right part is the similarity map between them with the most fine-grained copied segment annotations shown as black boxes. There are multiple boxes in frame-to-frame similarity map which can also be merged, and metric should not change dramatically whether these segment pairs are merged or not. These two similarity maps are quite different in appearance due to video duration and self-similarity between frames.



(c)	
Segment Recall	1.0
Segment Precision	1.0
Frame Recall	≈0.1
Frame Precision	≈0.1
Our Precision	≈0.1
Our Recall	≈0.1

(d)	
Segment Recall	1.0
Segment Precision	0.5
Frame Recall	≈1.0
Frame Precision	≈1.0
Our Precision	≈1.0
Our Recall	≈1.0

(e)	
Segment Recall	0.5
Segment Precision	1.0
Frame Recall	≈1.0
Frame Precision	≈1.0
Our Precision	≈1.0
Our Recall	≈1.0

(f)	
Segment Recall	0.0
Segment Precision	0.0
Frame Recall	≈1.0
Frame Precision	≈1.0
Our Precision	0.0
Our Recall	0.0

Figure 4. (a-b) illustrates calculation process of our proposed methods. (c-f) provide four simplified cases for comparison between our metric and extended segment and frame metrics. The gray dashed lines in (c-f) represent temporal sequential copy, and other common cases including random order video editing can also happen with sophisticated pattern in similarity map.

and then obtain the union frames $L^x_{O_j}$ and $L^y_{O_j}$ of all GT boxes for each predicted box. The precision is defined in a similar way as the recall:

$$Precision = \frac{\sum_{j=0}^{n} L^x_{O_j}}{\sum_{j=0}^{n} L^x_{P_j}} \cdot \frac{\sum_{j=0}^{n} L^y_{O_j}}{\sum_{j=0}^{n} L^y_{P_j}} \tag{6}$$

, where $L^x_{P_j}$ and $L^y_{P_j}$ are the width and height of predicted box P_j respectively.

To calculate the final score, the harmonic mean of recall and precision, i.e. $Fscore$, is adopted as the primary metric:

$$Fscore = \frac{2 \cdot Recall \cdot Precision}{Recall + Precision} \tag{7}$$

4.3. Comparison

Technically, previous segment and frame P/R protocol cannot be used for video pairs as input. To make them comparable to the proposed metric in this scenario, we extend them by simply calculating the metrics on both x and y axis respectively. Fig.4(c-f) provide several simplified and extreme cases with the evaluation results under different metrics. As shown in Fig.4(c), extended segment P/R metric cannot reflect the inaccurate boundary of the predicted copied segments. For multiple sub-segments emphasized in Sec.4.1, previous segment metric also lacks robustness indicated in Fig.4(d-e). Frame P/R shows poor measurements in the case shown in Fig.4(f) where predicted copied segments totally misalign with GT segments.

By calculating the frame-level prediction accuracy inside each segment pair, our proposed metric shows better adaptability for different video copy scenarios. With IoU of the segment pair (represented as bounding box in similarity map) taken into account, the temporal correlation between two videos is highlighted in our metric. This overcomes the obvious drawback of frame-level statistic mentioned before but still maintains the accuracy measurement with fine granularity. Meanwhile, our metric is also robust for segment division that significantly affects previous segment metric. We also discuss the limitation of our proposed metric on some rare and extreme cases in Sec.S6 of Supplementary Material. In addition, mAP from action localization task [30, 31] is not suitable for copy localization. The core of mAP is to inspect the temporal (1D) IoU with the GT segment with evaluated result of true or false positive. However, the pair of input videos in video copy scenarios both might contain multiple partial copies, and the proposed metric should better measure prediction accuracy of copy overlapping segments between a video pair, rather than evaluating 1D temporal localization for a single video.

5. Benchmark

5.1. Pipeline

We outline our pipeline in Fig.5. The pipeline starts with a pair of potential copied videos as input and then outputs the predicted copied segments.

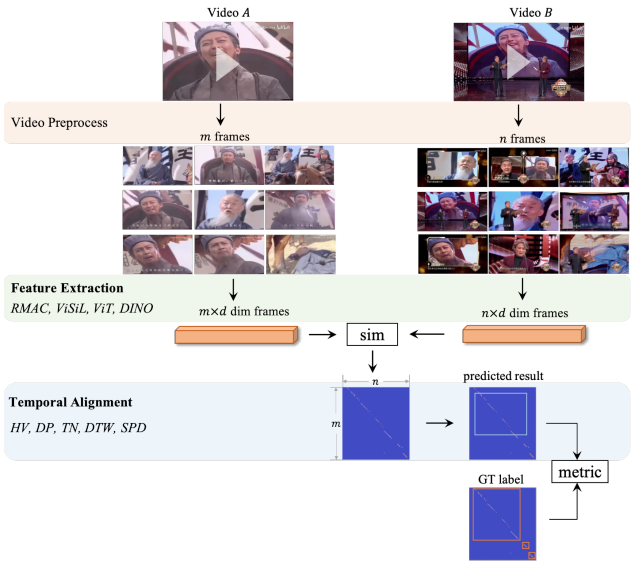


Figure 5. The benchmark pipeline.

The first step is the video preprocessing including video decoding and frame extraction. After this step, the input video is represented as a sequence of frames for subsequent process. Here, the frame extraction can be uniform or non-uniform based on keyframe selection or video summarization algorithms [32, 33]. As this part is beyond the scope of this paper, we simply adopt the commonly used uniform sampling strategy.

After obtaining a set of frames (e.g., m frames), we need to employ a visual descriptor on these m frames and construct a $m \times d$ dimension embedding for each video. d is the feature dimension for each frame. Then a frame-to-frame similarity map can be constructed by calculating the frame-to-frame similarity between the two video embeddings. In this paper, we select the following four feature extractors for comparison: classical and commonly used R-MAC feature [18]; ViSiL feature that is proved SOTA on video-level retrieval task [5] but with high dimension; ViT that shows comparable performance on image retrieval task [34]; recently self-supervised pretrained model DINO [23] with competitive results on image copy detection. For R-MAC, ViT and DINO, similarity matrices are calculated by cosine similarity, and the ones for ViSiL feature are calculated by Chamfer Similarity which is consistent with [5].

It is notable that all the selected feature extractors are frame-level. Meanwhile, there are methods [35, 36] can jointly learn spatial-temporal features. However, in our segment-level video copy detection task, the copy overlap boundaries should be precisely localized, hence the spatial-temporal features extracted on video clips with a minimum length of several seconds are not appropriate. But temporal info can be utilized after frame feature extraction and show improved performance on video retrieval [17]. We believe that both temporal and spatial attention can be flexibly inserted into the pipeline and the entire algorithm process could be trained in an end-to-end manner, and we leave this to our future work.

The final step is to find temporal alignments and output the copied segment pairs. As mentioned in Sec.4, the predicted results can be represented as bounding boxes in similarity map shown in Fig.5. Here, we re-implement the following five alignment methods: Hough Voting (HV) [24], Temporal Network (TN) [25], Dynamic Programming (DP) [26], dynamic time warping (DTW) [37] and recent SPD [8]. The previous three methods are commonly used for video copy detection [11] and video retrieval task comparison [5]. DTW is usually adopted to match two time sequence, and we simply modify it for adaption to our task. SPD shows SOTA temporal alignment results on VCDB with the previous segment-level metric.

5.2. Implementation Details

Dataset: For the purpose of training and evaluation for benchmark and future works, we split VCSL dataset with at least one query set in train/val/test for each topic category. The query sets and copied video pairs in train/val/test are 60/32/30 and 97712/42031/27765 respectively. The training, validation and test sets contain different sets of videos, providing completely independent contents. The detailed dataset split is given along with VCSL dataset release.

Metric: In Sec.4, we give the detailed evaluation metric for a copied video pair. For VCSL test datasets, we need to obtain an overall result on these 27765 pairs. Directly averaging all the metric results is inappropriate, since the video pair number of each query set is unbalanced. This calculation skews the overall results towards the query set with more data, making the diversity of query sets useless. Therefore, we first obtain the average metric inside each query set, and then calculate the macro average over query set as the overall result.

Feature extraction: We extract one frame per second for each video. All models for frame feature are pretrained on ImageNet [38] without other external datasets.

Temporal alignment: Since all five temporal alignment algorithms are traditional methods without training process except for SPD, we tune their hyper parameters on the validation set of VCLS and compare the results on the test set. For SPD, we train two versions of the network respectively on VCDB and train/val set of VCLS, and also evaluate them

Table 2. Comparison of benchmarked approaches with different features and alignment methods. The footnote in SPD ($_1$) denotes SPD trained on VCDB. ($_2$) denotes SPD trained on train/val set of VCSL. More fine-grained results are given in Sec.S3-S5 of Supplementary Material.

macro over Query Set		Recall	Precision	**F-score**
R-MAC 512 dim.	HV	77.65	75.08	76.34
	TN	82.05	87.95	84.90
	DP	61.04	87.37	71.87
	DTW	55.10	85.74	67.09
	SPD_1	79.39	91.37	84.96
	SPD_2	82.16	89.79	**85.81**
ViSiL 9*3840 dim.	HV	81.93	71.64	76.44
	TN	82.16	89.56	85.70
	DP	64.28	89.76	74.91
	DTW	54.27	91.40	68.10
	SPD_1	79.16	91.45	84.86
	SPD_2	83.87	88.97	**86.34**
ViT 768 dim.	HV	76.71	75.70	76.20
	TN	83.60	86.22	84.89
	DP	60.61	81.20	69.41
	DTW	55.40	72.99	62.99
	SPD_1	80.56	90.28	85.14
	SPD_2	81.61	90.94	**86.02**
DINO 1536 dim.	HV	81.46	73.17	77.09
	TN	88.74	83.69	86.14
	DP	64.36	86.58	73.83
	DTW	57.16	84.92	68.33
	SPD_1	81.23	90.66	85.69
	SPD_2	84.67	90.31	**87.40**

on the test set of VCLS.

Other detailed experimental settings can be inferred in our released benchmark codes.

5.3. Results and Discussion

Table 2 presents the performance of all possible combinations of the features and alignment methods with our proposed metric.

As can be shown from Table 2, the frame feature does indeed have impacts on the performance, and there shows similar trend with different alignment methods. DINO achieves the best results with a moderate size of feature dimension, which might be attributed to the transformer architecture and the self-supervised framework. However, the effect of features on the final results is not as dramatic as expected, especially considering different feature dimensions. By observing the similarity map, it can be found that the pattern of copied segments is not obvious for some hard cases (picture in picture from variety shows, crop in a large margin from kichiku), and all methods show poor performance. Similarity maps on some example hard cases are given in Sec.S3 of Supplementary Material. We suspect that it is due to the limitation of global features that can not capture local correspondences between severely transformed frames. We hope that the hard cases in VCSL give some insights to develop more powerful feature representations for the segment-level video copy detection task.

In the aspect of temporal alignment methods, SPD trained on the training set of VCSL performs the best. It is notable that SPD results trained on VCDB dataset are even lower than TN with some features. This indicates the importance of large-scale and well-annotated datasets, especially for supervised learning methods. Among all the combination of methods, DINO + SPD_2 performs best with an F-score of over 87%. However, results on some specific query sets are only around 50% which are far from satisfactory, especially on some query sets in kichiku and movie category with significant temporal and spatial editing. Such recently emerged copy infringement types in VCSL bring great challenges to these temporal alignment methods designed for near duplicated cases. The details and bad cases analysis are given in Sec.S4 of Supplementary Material.

Besides the overall results above, we also evaluate the algorithms at different data distribution with finer granularity. Corresponding to Fig.2, F-score performance results on video duration, segment duration, segment number per video pair, copy duration percentage are indicated in Sec.S5 of Supplementary Material with detailed analysis. Video pairs containing more segment copies and lower copy duration percentage meet significantly more difficulties with lower results. VCSL provides a large amount of these types to motivate future algorithm evolution. In addition, we can also observe that temporal alignment methods show different adaptability on various data distribution and situations, e.g., SPD shows better performance on backwards-running videos, and TN is more suitable for multiple copied segments per video pair. This is due to their different definition and constraints on copy detection task. Fine-grained video copy detection and consequent model fusion might also be opportunities for future research.

6. Conclusion

This work represents the currently largest segment-level video copy detection dataset, VCSL. Compared with the existing partial copy detection dataset (VCDB), VCSL has two orders of magnitude more labelled data, and is collected from realistic, challenging YouTube and Bilibili videos. In addition, we refine the evaluation protocol and jointly propose a new metric to address existing problems revealed by the previous evaluation protocol and metric. Four feature extraction methods and five temporal alignment methods are quantitatively evaluated and compared, which reveals interesting future research directions. We hope that the public availability of VCSL and new thoughtful metric will motivate even more interest in this important and applicable field for video copy detection and copyright protection.

References

[1] Statistics from youtube website. http://www.youtube.com/yt/press/statistics.html/, 2021. 1

[2] Chinese video platform bilibili mau up 38% in q2 2021. http://www.chinainternetwatch.com/31131/bilibili-quarterly/, 2021. 1

[3] Yi Xu, true Price, Fabian Monrose, and Jan-Michael Frahm. Caught red-handed: Toward practical video-based subsequences matching in the presence of real-world transformations. In *2017 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pages 1397–1406, 2017. 1

[4] Julien Law-To, Li Chen, Alexis Joly, Ivan Laptev, Olivier Buisson, Valerie Gouet-Brunet, Nozha Boujemaa, and Fred Stentiford. Video copy detection: a comparative study. In *Proceedings of the 6th ACM international conference on Image and video retrieval*, pages 371–378, 2007. 1

[5] Giorgos Kordopatis-Zilos, Symeon Papadopoulos, Ioannis Patras, and Ioannis Kompatsiaris. Visil: Fine-grained spatio-temporal video similarity learning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 6351–6360, 2019. 1, 2, 3, 7

[6] Lowik Chanussot, Filip Radenovic, Tomas Jenicek, Maxim Maximov, Laura Leal-Taixé, Ismail Elezi, Ondrej Chum, and Cristian Canton Ferrer. The 2021 image similarity dataset and challenge. 1

[7] L. Baraldi, M. Douze, R. Cucchiara, and H. Jegou. Lamv: Learning to align and match videos with kernelized temporal layers. In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7804–7813, 2018. 1, 2, 3

[8] Chen Jiang, Kaiming Huang, Sifeng He, Xudong Yang, Wei Zhang, Xiaobo Zhang, Yuan Cheng, Lei Yang, Qing Wang, Furong Xu, et al. Learning segment similarity and alignment in large-scale content based video retrieval. In *Proceedings of the 29th ACM International Conference on Multimedia*, pages 1618–1626, 2021. 1, 2, 3, 4, 7

[9] 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*, pages 4165–4173, 2021. 1, 2

[10] Weijun Tan, Hongwei Guo, and Rushuai Liu. A fast partial video copy detection using knn and global feature database. *arXiv preprint arXiv:2105.01713*, 2021. 1

[11] 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*, pages 357–371. Springer, 2014. 1, 2, 3, 4, 7

[12] 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*, pages 218–227, 2007. 1, 2

[13] Giorgos Kordopatis-Zilos, Symeon Papadopoulos, Ioannis Patras, and Ioannis Kompatsiaris. Fivr: Fine-grained incident video retrieval. *IEEE Transactions on Multimedia*, 21(10):2638–2652, 2019. 1, 2

[14] 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 International Conference on Computer Vision*, 2019. 1, 2

[15] J. Law-To, A. Joly, and N. Boujemaa. Muscle-vcd-2007: a live benchmark for video copy detection. 2007. 1, 2

[16] W. Kraaij and G. Awad. Trecvid 2011 content-based copy detection: Task overview. *Online Proceedings of TRECVid 2010*, 2011. 1, 2

[17] 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 (WACV)*, January 2021. 3, 7

[18] Albert Gordo, Jon Almazan, Jerome Revaud, and Diane Larlus. End-to-end learning of deep visual representations for image retrieval. *International Journal of Computer Vision*, 124(2):237–254, 2017. 3, 7

[19] Giorgos Kordopatis-Zilos, Symeon Papadopoulos, Ioannis Patras, and Yiannis Kompatsiaris. Near-duplicate video retrieval with deep metric learning. In *IEEE International Conference on Computer Vision Workshop (ICCVW)*, 2017. 3

[20] 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*, 2017. 3

[21] 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*, pages 5998–6008, 2017. 3

[22] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020. 3

[23] Mathilde Caron, Hugo Touvron, Ishan Misra, Hervé Jégou, Julien Mairal, Piotr Bojanowski, and Armand Joulin. Emerging properties in self-supervised vision transformers. *arXiv preprint arXiv:2104.14294*, 2021. 3, 7

[24] M. Douze, H. Jegou, and C. Schmid. An image-based approach to video copy detection with spatio-temporal post-filtering. *IEEE Transactions on Multimedia*, 12(4):257–266, 2010. 3, 7

[25] Hung-Khoon Tan, Chong-Wah Ngo, Richard Hong, and Tat-Seng Chua. Scalable detection of partial near-duplicate videos by visual-temporal consistency. MM '09, page 145–154, New York, NY, USA, 2009. Association for Computing Machinery. 3, 7

[26] C. Chou, H. Chen, and S. Lee. Pattern-based near-duplicate video retrieval and localization on web-scale videos. *IEEE Transactions on Multimedia*, 17(3):382–395, 2015. 3, 7

[27] Sebastien Poullot, Shunsuke Tsukatani, Anh Nguyen, Hervé Jégou, and Shin'ichi Satoh. Temporal matching kernel with explicit feature maps. 10 2015. 3

[28] Md Atiqur Rahman and Yang Wang. Optimizing intersection-over-union in deep neural networks for image segmentation. In *International symposium on visual computing*, pages 234–244. Springer, 2016. 5

[29] Hamid Rezatofighi, Nathan Tsoi, JunYoung Gwak, Amir Sadeghian, Ian Reid, and Silvio Savarese. Generalized intersection over union: A metric and a loss for bounding box regression. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 658–666, 2019. 5

[30] Fabian Caba Heilbron, Victor Escorcia, Bernard Ghanem, and Juan Carlos Niebles. Activitynet: A large-scale video benchmark for human activity understanding. In *Proceedings of the ieee conference on computer vision and pattern recognition*, pages 961–970, 2015. 6

[31] Will Kay, Joao Carreira, Karen Simonyan, Brian Zhang, Chloe Hillier, Sudheendra Vijayanarasimhan, Fabio Viola, Tim Green, Trevor Back, Paul Natsev, et al. The kinetics human action video dataset. *arXiv preprint arXiv:1705.06950*, 2017. 6

[32] Evlampios Apostolidis, Eleni Adamantidou, Alexandros I Metsai, Vasileios Mezaris, and Ioannis Patras. Video summarization using deep neural networks: A survey. *arXiv preprint arXiv:2101.06072*, 2021. 7

[33] Wencheng Zhu, Jiwen Lu, Jiahao Li, and Jie Zhou. Dsnet: A flexible detect-to-summarize network for video summarization. *IEEE Transactions on Image Processing*, 30:948–962, 2020. 7

[34] Alaaeldin El-Nouby, Natalia Neverova, Ivan Laptev, and Hervé Jégou. Training vision transformers for image retrieval. *arXiv preprint arXiv:2102.05644*, 2021. 7

[35] 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*, pages 6299–6308, 2017. 7

[36] Du Tran, Lubomir Bourdev, Rob Fergus, Lorenzo Torresani, and Manohar Paluri. Learning spatiotemporal features with 3d convolutional networks. In *Proceedings of the IEEE international conference on computer vision*, pages 4489–4497, 2015. 7

[37] Donald J Berndt and James Clifford. Using dynamic time warping to find patterns in time series. In *KDD workshop*, volume 10, pages 359–370. Seattle, WA, USA:, 1994. 7

[38] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. *Advances in neural information processing systems*, 25:1097–1105, 2012. 7