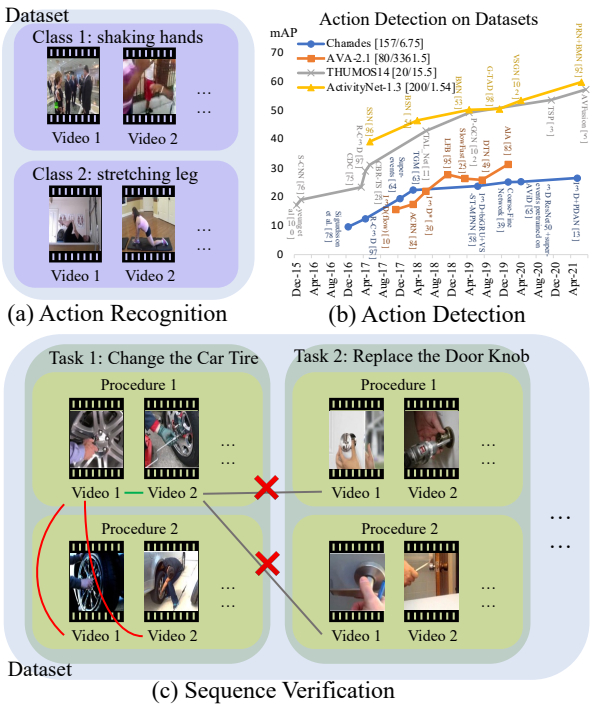
SVIP: Sequence VerIfication for Procedures in Videos

Yicheng Qian[1], Weixin Luo[2], Dongze Lian[1,5], Xu Tang[3], Peilin Zhao[4], Shenghua Gao[1,6,7†]

[1]ShanghaiTech University, [2]Meituan, [3]Xiaohongshu Inc., [4]Tencent AI Lab
[5]National University of Singapore, [6]Engineering Research Center of Intelligent Vision and Imaging
[7]Shanghai Engineering Research Center of Energy Efficient and Custom AI IC
{qianych, luowx, liandz, gaoshh}@shanghaitech.edu.cn,
tangshen@xiaohongshu.com, masonzhao@tencent.com

Abstract

In this paper, we propose a novel sequence verification task that aims to distinguish positive video pairs performing the same action sequence from negative ones with step-level transformations but still conducting the same task. Such a challenging task resides in an open-set setting without prior action detection or segmentation that requires event-level or even frame-level annotations. To that end, we carefully reorganize two publicly available action-related datasets with step-procedure-task structure. To fully investigate the effectiveness of any method, we collect a scripted video dataset enumerating all kinds of step-level transformations in chemical experiments. Besides, a novel evaluation metric Weighted Distance Ratio is introduced to ensure equivalence for different step-level transformations during evaluation. In the end, a simple but effective baseline based on the transformer encoder with a novel sequence alignment loss is introduced to better characterize long-term dependency between steps, which outperforms other action recognition methods. Codes and data will be released[1].



(a) Action Recognition (b) Action Detection

(c) Sequence Verification

Figure 1. Comparison between traditional action tasks and sequence verification. **(a)** Action recognition datasets generally consist of various action categories containing videos; **(b)** Results of action detection methods on different datasets in recent years. $[a/b]$ means the corresponding dataset contains a action classes and b action instances per video; **(c)** Sequence verification dataset aims to verify the procedures in the same task. Cross-task verification is dropped due to its simplicity.

1. Introduction

In recent years, short-form videos filming people's daily life widely disseminated on social media, which leads to the spurt of activity videos and greatly facilitated the research on video understanding [6, 14, 32, 55, 69, 85] as well. One can see from these videos that most daily activities are accomplished by serial steps instead of a single step. Such sequential steps form a procedure of which key-steps obey intrinsic consistency, while different participants may accomplish the same activity by different procedures with step-level divergence, as shown in Figure 1 (c). In this paper, we advocate a novel action task **sequence verification** which intends to verify whether the procedures in two videos are step-level consistent, which can be applied to multiple potential tasks such as instructional training and performance

† Corresponding Author

[1]https://github.com/svip-lab/SVIP-Sequence-VerIfication-for-Procedures-in-Videos

scoring. To better demonstrate this task, we define related terms specifically. *Step:* a human-action or human-object-interaction atomic unit that is always labeled by a verb, a noun, and even prepositions, *e.g.*, 'remove the old wrapper'; *procedure:* a sequence of steps performed in the chronological order to accomplish a certain goal, *e.g.*, 'remove the old wrapper - wrap with the new wrapper'; *task:* an activity that needs to be accomplished within a defined period of time or by a deadline, *e.g.*, 'Rewrap battery' and 'Change the car tire' in COIN. We note that a task can be accomplished by various procedures; *video:* each video performs one procedure of a certain task; *P/N pairs:* two videos performing an identical procedure form a positive pair, while those performing different procedures from the same task form a negative pair.

Why do we need sequence verification? Traditional action tasks such as action recognition [52, 90, 104], action localization [11, 50, 75] and action segmentation [24, 47, 100] have achieved significant progress due to the development of CNN as well as recently prevalent visual transformer [20]. However, most of these tasks follow a close-set setting with a limitation of predefined categories, illustrated in Figure 1 (a). Besides, accurate annotations of steps in numerous videos are extremely time-consuming and labor-intensive, followed by boundary ambiguities that have been studied in recent work [46, 72, 85, 107] though. However, our proposed sequence verification task circumvents both of these problems by verifying any video pair according to their distance in embedding space. In this way, the sequence verification task neither requires the predefined labels nor consumes the intensive step annotations, which can easily handle the open-set setting.

As shown in Figure 1 (c), our proposed sequence verification aims to verify those procedures with semantic-similar steps rather than being associated with totally irrelevant tasks, which enables it to concentrate more on action step association rather than background distinction. Thus, an appropriate dataset is crucial to perform this task well. However, existing trimmed video datasets such as UCF101 [82], Kinetics [10], and Moments in Time [59] etc. are leveraged to carry out single-label action recognition. On the other hand, untrimmed video datasets like EPIC-KITCHENS [15], Breakfast [43], Hollywood Extended [7], ActivityNet [8] provide videos composed by multiple sub-actions and the corresponding step annotations, but they do not collect videos that especially performs similar or identical procedures. Thus, they cannot be used directly for sequence verification. To this end, we rearrange some datasets such as COIN [85] and Diving48 [51] where each task contains multiple videos recording different procedures, and each video has step-level annotations. Generally, videos with the same procedure are assigned to an individual category for training. Positive pairs and negative ones for testing are collected within the same procedure and cross different procedures in the same task, respectively. It should be noticed that these unscripted videos in the same procedure could be with a large appearance variance due to background divergence and personal preference, which makes sequence verification more challenging. Apart from that, we introduce a scripted filming dataset performing chemical procedures, where it includes all kinds of step-level transformations such as deletions, additions, and order exchanges. Thus, the effectiveness of any algorithm can be well justified by this newly proposed dataset. Additionally, since more step transformations may lead to larger feature distances which is unfair compared to less ones, we introduce a new evaluation metric Weighted Distance Ratio to make sure that every negative pair will be counted equally regardless of its step-level difference during evaluation.

As an unprecedented task, sequence verification may be solved by off-the-shelf action detectors [10, 13, 25, 30, 39, 49, 57, 62–64, 77, 83, 84, 94, 96]. However, their performance on Charades [78] or AVA [32], shown in Figure 1 (b), is not satisfactory to conduct step-level detection before verification. Although [3, 5, 11, 28, 53, 54, 74, 75, 91, 95–97, 99, 101, 102] perform well on ActivityNet [8] or THUMOS14 [38], it lacks persuasion since the two datasets either contain a few action classes or action instances per video. Thus, we introduce a simple but effective baseline CosAlignment Transformer (abbreviated as CAT), which leverages 2D convolution to extract discriminative features from sampled frames and utilizes a transformer encoder to model inter-step temporal correlation in a video clip. Whereas representing the whole video with multiple steps as a single feature vector may lose information corresponding to the order of steps in a procedure. Thus, we introduce a sequence alignment loss that aligns each step in a positive video pair via the cosine similarities between two videos. The results show that our proposed method significantly outperforms other action recognition methods in the sequence verification task.

We summarize our contributions as follows:

i) **Problem setting:** We propose a new task, sequence verification. To our knowledge, this is the first task focusing on procedure-level verification between videos.

ii) **Benchmark:** We rearrange two unscripted video datasets with significant diversity and propose a new scripted dataset with multiple step-level transformations to support this task. Moreover, a new evaluation metric is introduced especially for this novel task.

iii) **Technical contributions:** We propose a simple but effective baseline that contains a transformer encoder to explicitly model the correlations between steps. Besides, a sequence alignment loss is introduced to improve the sensitivity to step disorder and absence. This novel baseline significantly outperforms other action recognition methods.

Dataset	# Tasks	# Videos	# Steps	# Procedures	# Split Videos	# Split Samples
COIN-SV	36	2,114	749	37 / 268 / 285	1,221 / 451 / 442	21,741 / 1,000 / 400
Diving48-SV	1	16,997	24	20 / 20 / 8	6,035 / 7,938 / 3,024	50,000 / 1,000 / 400
CSV	14	1,940	106	45 / 25 / -	901 / 1,039 / -	8,531 / 1,000 / -

Table 1. The statistical information of three datasets. It is listed with an order of training, testing, and validation.

2. Related Work

Action Tasks. Traditional action-related tasks such as action recognition, action detection, and action segmentation have been greatly developed due to the advances in CNNs. i) As a means of general video representation, deep-learning-based **action recognition** can be generally summarized to stream-based methods [10, 12, 17, 19, 26, 52, 65, 79, 87, 88, 90, 104] and skeleton-based [21, 80, 92, 98] methods. Both kinds of methods aim to produce a feature representation for each trimmed video, to which a video-level label over predefined action categories is predicted according. ii) To seek the interested sub-actions in untrimmed videos, **action detection** [13, 25, 39, 53, 54, 57, 62–64, 83, 96, 103] are proposed to detect the start and end of sub-action instances and predict their categories. iii) To conduct dense action predictions in untrimmed videos, **action segmentation** is designed to label each frame including background in videos. With dense annotations, fully-supervised methods [23, 48, 58, 66, 68, 73, 75, 99] rely on sliding windows, Markov models, or temporal convolutional networks to model the temporal relations. However, dense annotations in videos require expensive human-labors as well as consume much time, though the weakly-supervised methods [7, 18, 36, 45, 67] with order labels of actions only have achieved satisfactory performance. Last but not least, it still remains a concern if these action-related tasks are able to generalize well to unknown classes in the wild. Different from them, our task has no restriction during inference, so it can easily tackle an open-set setting.

Video datasets. Multiple existing video datasets [1, 8, 15, 31, 40, 41, 44, 60, 78, 82] have being dominated on video understanding during a long period. To begin with, HMDB51 [44] and UCF101 [82] that contains 51 and 101 classes of actions, respectively, are introduced for action recognition. Next, Something-Something [31] collects 147 classes of interactions between humans and objects in daily life. In addition, ActivityNet [8] and Kinetics [41] collect videos from YouTube and builds large-scale action recognition datasets. Other datasets for instructional video summarization and analysis [68, 81, 85] that are annotated with texts and temporal boundaries of a series of steps, contributes to the understanding of language and vision. EPIC-KITCHENS dataset [15] collects the human actions such as washing glass or cutting bell pepper in kitchen scenes and targets at the first-person perspective to reflect people's goals and motivations.

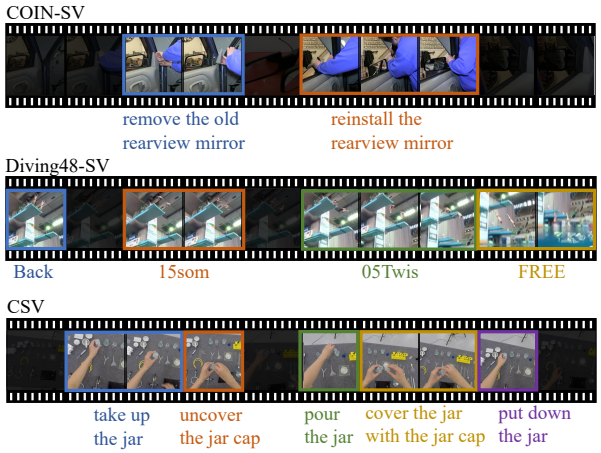


Figure 2. Dataset illustration. COIN-SV contains 36 daily life tasks such as 'Replace rearview mirror glass', 'Make burger', leading to high background diversity. Diving48-SV and CSV consist of videos in the diving competition and chemical experiments scene, respectively. Every video in three datasets is categorized by the sequence of steps it performs, as known as the procedure in the video. Note that only COIN-SV provides temporal annotation for steps, while Diving48-SV and CSV only provide the procedure-level annotation for each video.

Instructional videos analysis. Instructional videos are generally accompanied with explanations such as audio or narrations matching the timestamps of sequential actions which has attracted the research interest in the video understanding community. For instance, step localization [58, 85, 105] as well as action segmentation [27, 29, 61, 85] in instructional videos have been widely studied in the early stage. With the growing attention paid to this research topic, various kinds of tasks related to instructional videos have been proposed, *e.g.*, video captioning [37, 56, 86, 105] which generates the description of a video based on the actions and events, visual grounding [35, 76] which locates the target in an image according to the language description, and procedure learning [2, 22, 27, 71, 72, 105] which extracts key-steps.

3. Data Preparation

Due to the intrinsic step-procedure-task structure in the publicly available datasets COIN [85], and Diving48 [51], we reorganize these two datasets to support our proposed sequence verification task focusing on verifying various

step-level transformations. However, the procedures in the same task may lack enough diversity in these datasets to fully verify the effectiveness of our proposed method for sequence verification. Thus, we collect a novel scripted dataset, Chemical Sequence Verification, enumerating all kinds of procedures in the same task, which will be introduced later. The statistics of these three datasets can be found in Table 1. We visualize some samples in Figure 2.

The rest of this section introduces the common structure, specific processing and basic information of these datasets.

3.1. Common Structure

As shown in Figure 1 (c), each dataset used in this paper contains videos completing various tasks, *e.g.*, the original COIN dataset contains 180 tasks common in daily life. In practice, each individual task can be accomplished by different procedures of which steps as atomic actions still obey certain orders. Meanwhile, the steps of two procedures with different task-orientations will not overlap each other at most times. Thus, we will not introduce sequence verification cross tasks to ensure the challenge.

3.2. COIN-SV

COIN [85] is a comprehensive instructional video dataset that contains 180 tasks such as 'Replace the door knob', 'Change the car tire', and 'Install a ceiling fan'. This recently proposed dataset is quite challenging due to its background diversity and even significant distinctions between videos of the same procedures, which benefits our proposed sequence verification task. In total, it contains 11827 videos over 4715 procedures, which means COIN is followed by a long-tail distribution where most procedures have one or two videos only. To facilitate the classifier training, we preserve 36 tasks that contains at least one procedure with more than 20 videos and discard the other tasks. Procedures with more than 20 videos are used for training, and the rest are assigned to the validation and testing sets randomly. Since the original split in this dataset is reorganized, we name it COIN-SV.

3.3. Diving48-SV

Diving48 [51] dataset records diving competition videos with 48 kinds of diving procedures standardized by the international federation FINA, which consists of around 18,000 trimmed videos. Each diving procedure is a sub-action sequence of one-step takeoff, two-step movements in flight, and one-step entry. In total, 16997 videos over 48 procedures are publicly available up to now. Obviously, this dataset is less challenging than COIN due to its dual background including a board, a pool, and spectators and less step-level divergence. We assign 20, 8, 20 procedures for the training, validation, and testing sets, respectively. Similar to COIN-SV, we name it Diving48-SV.

3.4. Chemical Sequence Verification

Since the videos in COIN-SV and Diving48-SV are gathered from the internet, it is difficult to include all kinds of step-level transformations without predefined scripts, which is crucial for the sequence verification task. To this end, we collect a new dataset named Chemical Sequence Verification (CSV) containing videos with all kinds of step-level transformations such as deletions, additions, and order exchanges. Concretely, volunteers from an egocentric perspective are asked to conduct chemical experiments with predefined scripts. In a word, the CSV dataset includes 14 tasks, and each consists of 5 procedures. We select 45 procedures for training and 25 procedures for testing. CSV has no validation set due to its limited number of procedures/videos. Data gathering process, video annotations, and statistics information are available in the supplementary material.

4. Method

Classical models for video action recognition [10, 12, 17, 19, 26, 65, 79, 87, 88, 90] aim to predict action categories without paying attention to sub-action orders as many as possible due to simple frame feature aggregation such as pooling. Nevertheless, our task intends to verify two videos with large as well as subtle step-level transformations. For instance, a video performing A, then B, and finally C is treated as a negative sample to another video carrying out A and finally C, while both of them may be successfully predicted by a traditional action classifier. To fit our proposed task, we introduce two remedies over the traditional action classification during training: i) procedures rather than tasks in the training set are regarded as training classes, in order to enable the model to distinguish those procedures even with tiny step-level transformations in the same task; ii) since pooling over frame features may bring order insensitivity to the model, we remain the temporal dimension without any down-sampling operation and it is finally reshaped to the channel dimension, followed by a fully-connected layer with order-sensitivity.

4.1. Preliminary

For a certain dataset $D = \{(V_i, S_i)\}_{i=1}^{n}$, a set of n video clips V are given with corresponding procedure annotations S. **Here we do not use the timestamp annotations of steps since action detectors will not be used in this paper.** We denote the model as $f : \mathbb{R}^{3\times H\times W\times K} \rightarrow \mathbb{R}^{C}$. K is the number of sampled frames in a video. H and W are frames' height and width, respectively. C is the total number of procedures in the training set. Following the paradigm in face verification [4, 16, 70], we treat sequence verification as a multi-category classification task during training, and videos performing the same procedure

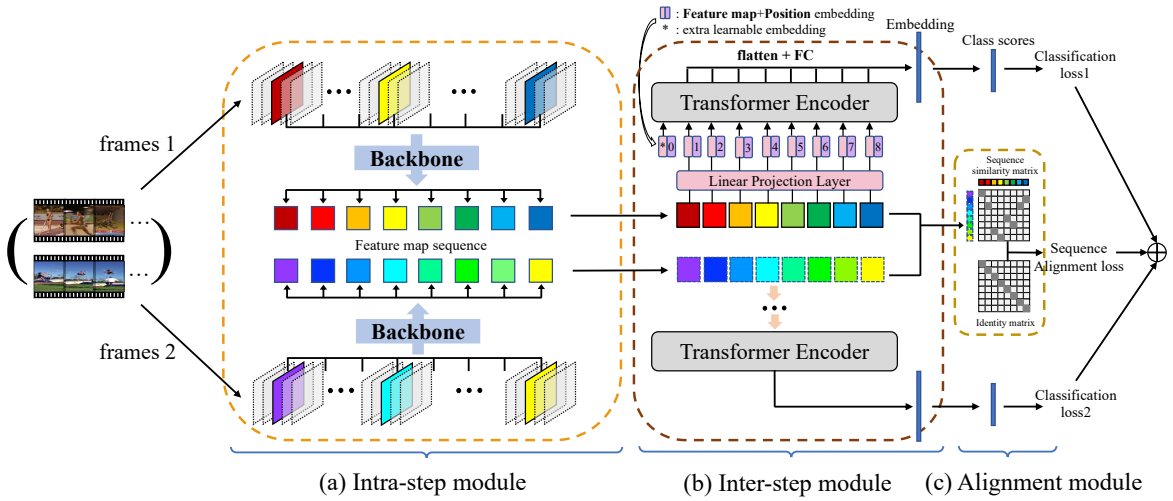


Figure 3. Pipeline overview. **a)** Applying a 2D backbone on the sampled frames (colored in the input frames) to capture features of individual steps, which we called intra-step features. The output of this module is a sequence of feature maps that are temporally modeled from the sampled frames. **b)** Applying a transformer encoder to aggregate the sequential feature maps. **c)** This module aims at imposing the two sequences of feature maps to be matched.

are classified into the same category. In the testing phase, we collect the videos from the same procedure to form positive pairs and the videos from different procedures but still in the same task to form negative pairs. Then embedding distance between two videos in a pair indicates the verification score of this pair. The procedure classification loss L_{cls} is as follows.

$$L_{\text{cls}} = \sum_{i=1}^{n} \delta(f(V_i), Y_i) \tag{1}$$

where δ is the cross-entropy function, Y_i is the C-dim one-hot vector whose entry corresponding to S_i is 1.

4.2. Baseline

We utilize a ResNet [34] backbone followed by a fully-connected layer to aggregate temporal information and a softmax classification layer as our baseline. Following TSN [90], we divide each input video into K segments ($K = 16$ in our experiments). One frame in each segment is randomly selected to form the input tensor $x \in \mathbb{R}^{3\times H\times W\times K}$, which is fed into the backbone and outputs a tensor with a shape of $D \times K$ where D is feature dimension. Then it is flattened into a vector for order-sensitivity. Finally, a procedure classifier with C categories is appended for training.

4.3. Transformer Encoder

Transformer [89] has achieved great success in Natural Language Process and it has been applied in multiple computer vision tasks such as image recognition [20, 93] and object detection [9, 106]. To better characterize inter-step correlations, we follow [20] and integrate the transformer encoder into the backbone by replacing the global average pooling. As Figure 3 describes, we firstly flatten the spatial feature maps and apply a trainable linear projection layer on these flattened vectors, resulting in feature vectors $\mathbf{E}_i \in \mathbb{R}^D, i = 1, 2, \cdots, K$. Further, randomly initialized position embedding is added to retain order information. In conclusion, the input $\mathbf{I}$ of a standard transformer encoder is

$$\mathbf{I} = [\mathbf{E}_1; \mathbf{E}_2; \cdots; \mathbf{E}_K] + \mathbf{E}_{pos}, \in \mathbb{R}^{K\times D}. \tag{2}$$

The output $\mathbf{O} \in \mathbb{R}^{K\times 1024}$ is flattened and fed into a fully-connected layer for a global representation of the input video. We adopt the sequential features instead of the CLS token since the former explicitly remains order information.

4.4. Sequence Alignment

So far, our proposed method aims to extract a global video representation supervised by the procedure classifier. However, the step order in a procedure is especially important in sequence verification. To make sure two positive procedures are step-level consistent, we propose a Sequence Alignment loss that explicitly imposes feature consistency step-by-step. Specifically, we extract the last spatial feature maps in the backbone and use global average pooling to produce feature vectors for all the frames in a given positive pair (seq_1, seq_2), where seq_i is the frame sequence sampled from video i. Then cosine similarity is calculated for all the frame pairs formed by the the two sequences, resulting in a correlation matrix:

$$\text{corr}_{ij} = \frac{f_{1i}}{\|f_{1i}\|} \cdot \frac{f_{2j}^T}{\|f_{2j}\|}, \tag{3}$$

where corr_{ij} denotes the similarity value at the i-th row and the j-th column of the matrix corr, while f_{1i} and f_{2j} represent the i-th feature of seq_1 and j-th feature of seq_2, respectively. Next, we perform a softmax function on each row of the similarity matrix to produce corr^1, whose i-th row is composed of cosine similarities between the i-th feature of seq_1 and every feature of seq_2. Similarly, we perform a softmax function on each column of the similarity and produce corr^2. We average these two matrices and denote the result as corr_{avg}. The diagonal values of corr_{avg} are then expected to be close to 1 while other values are expected to be close to 0, since both corr^1 and corr^2 have been normalized by softmax. In other words, we impose two videos in a positive pair to be similar in the feature space frame-by-frame, to some extent step-by-step. Mathematically, our proposed Sequence Alignment loss L_{seq} can be defined as:

$$L_{\text{seq}} = \|\mathbf{1} - h(\frac{\text{corr}^1 + \text{corr}^2}{2})\|_1, \tag{4}$$

where $\mathbf{1}$ is a vector whose entries are all one and h is a function to extract the diagonal entries of a matrix.

4.5. Training Loss

Now, we train the network by the procedure classification loss and the sequence alignment loss in an end-to-end manner. Thus, the total loss L can be summarized:

$$L = L_{\text{cls}} + \lambda L_{\text{seq}} \tag{5}$$

Here λ is a hyper-parameter and it sets to 1 by default.

4.6. Testing phase

During inference, the goal of sequence verification is to distinguish positive pairs from negative pairs. We denote each pair as $P_i = (V_{i_1}, V_{i_2})$. The model takes each video in P_j as input and produces one d-dimension visual embedding before the classification layer, which is denoted by $f' : \mathbb{R}^{K\times H\times W\times 3} \rightarrow \mathbb{R}^{D'}$. Next, we calculate the normalized Euclidean distance between the two procedures in the embedding space and the verification score y_i is defined:

$$d_i = g(f'(V_{i_1}), f'(V_{i_2})) \tag{6}$$

$$y_i = \begin{cases} 1, & d_i \leq \tau, \\ 0, & otherwise. \end{cases} \tag{7}$$

where g is a function that does l_2 normalization over two embeddings firstly and then calculates their Euclidean distance, τ is a threshold to decide whether the procedures are consistent. $y = 1$ means the procedures in two videos are consistent, otherwise inconsistent.

5. Experiments

5.1. Experimental Details

Datasets and setup. We conduct experiments on COIN-SV, Diving48-SV, and CSV. The specific information of each dataset is available in Section 3. Since this novel task is proposed to solve the open-set setting, there exists no procedure-level overlapping among the training, validation, and testing sets. However, step-level overlapping is unavoidable since different procedures can still contain several common steps.

Implementation Details. The ResNet-50 we employ is pre-trained on Kinetics-400 [41] to avoid over-fitting, while the new layers adopt Kaiming uniform initialization [33]. The experiments are conducted on 4 NVIDIA TITAN RTX GPUs with batch size 16, a cosine learning rate scheduler with a base learning rate of 0.0001, and weight decay 0.01. Adam [42] is used to optimize the whole network. For efficiency, we resize the raw images to 180×320. We also leverage horizontal flip, cropping, and color jittering for data augmentation. The feature dimension D' before the classifier layer is set to 128 for all experiments.

Baselines. Since we are the first to introduce the sequence verification task, there are no existing methods that are specially designed for this task. Considering that we learn video representation during training, which is similar to the action recognition task, we compare our proposed method with some advanced action recognition baselines: *Random*, *TSN [90]*, *TRN [104]*, *TSM [52]*, and *Video Swin [55]*.

Evaluation Metrics. (1) **AUC.** We adopt the Area Under ROC Curve (abbreviated as AUC) as one of the measurements, which is commonly used to evaluate the performance of face verification. Higher AUC denotes better performance. (2) **WDR.** It is short for Weighted Distance Ratio. To begin with, we calculate the mean embedding distance per unit Levenshtein distance for negative pairs in order to guarantee the equivalence of each pair during evaluation because larger step-level transformations always lead to larger embedding distance, discussed in Section 5.5. The mean embedding distance over positive pairs is then computed. In the end, we use the ratio between negative distance and positive distance, namely the Weighted Distance Ratio, as an indicator of performance for all methods. Obviously, its higher value means better performance the methods arrive at. Mathematically, we define WDR as:

$$\text{WDR} = \frac{\sum_{i=1}^{N} wd_i/N}{\sum_{j=1}^{P} d_j/P}, \tag{8}$$

where P and N are number of positives and negatives, repectively. d_i and d_j can be easily calculated by Equation 6. wd_i is defined as:

$$wd_i = \frac{d_i}{ed_i} \tag{9}$$

Method	Pretrain	#Param(M)	AUC / WDR				
			COIN-SV		Diving48-SV		CSV
			Val	Test	Val	Test	Test
Random	-	-	50.00 / -	50.00 / -	50.00 / -	50.00 / -	50.00 / -
TSN [90]	K-400	22.67	53.38 / 0.3651	47.01 / 0.3999	91.00 / 1.0835	81.87 / **0.6707**	59.85 / 0.3447
TRN [104]	K-400	23.74	54.92 / 0.3665	**57.19** / 0.3719	90.17 / **1.1438**	80.69 / 0.5876	80.32 / **0.4677**
TSM [52]	K-400	22.67	52.12 / 0.2948	51.25 / 0.3872	89.41 / 1.0035	78.19 / 0.5531	62.38 / 0.3308
Swin [55]	K-400	26.66	47.27 / 0.3895	43.70 / 0.3495	89.35 / 1.1066	73.10 / 0.5316	54.06 / 0.3141
CAT(ours)	K-400	72.32	**56.81** / **0.4005**	51.13 / **0.4098**	**91.91** / 1.0642	**83.11** / 0.6005	**83.02** / 0.4193

Table 2. Comparison with action recognition methods on the validation and testing set of COIN-SV, Diving48-SV, and CSV dataset.

Dataset	+TE	+SA	AUC (%)	WDR
COIN-SV			52.31	0.3677
	✓		55.46	0.3839
	✓	✓	**56.81**	**0.4005**
Diving48-SV			90.51	1.0093
	✓		90.91	1.0308
	✓	✓	**91.91**	**1.0642**
CSV			81.97	**0.4403**
	✓		82.07	0.4193
	✓	✓	**83.02**	0.4193

Table 3. Comparison between different model structures of our method on the testing set of the CSV dataset.

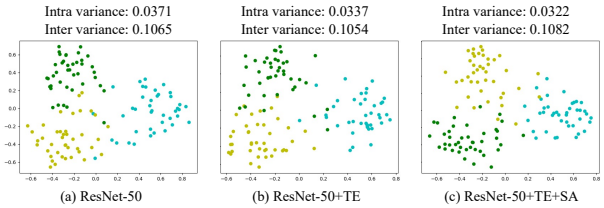


Figure 4. Visualizations of embeddings predicted by different model structures. Three kinds of colored points represent different procedures from the same task. The values above each sub-figure are the averaged intra-procedure variance and the inter-procedure variance, respectively.

where ed_i represents the text Levenshtein distance of pair i. Levenshtein distance, defined as *the minimum number of operations required to transform one string into the other* can be used as a measurement of how different in terms of steps two procedures are. More explanations and evaluations can be found in Section 5.5.

5.2. Comparison of Different Methods

The quantitative results of all the methods on the three datasets are shown in Table 2. We can find that our proposed CAT exceeds all other baselines in most cases, evaluated on the AUC metric. It is worth noticing that CAT does not achieve the best WDR in all datasets since the best model is selected by the highest AUC in the validation set. Apart from that, AUC on COIN-SV is extremely inferior compared to the other two datasets, which indicates its significant challenge due to its complex background and procedure diversity. Surprisingly, Video Swin Transformer is inferior to other baselines. We conjecture that it suffers from data insufficiency.

5.3. Ablation Study

In this section, we investigate the effectiveness of the transformer encoder (TE) and sequence alignment (SA) module. The experiments are conducted on the testing set of the CSV dataset if not specially stated. Specifically, we gradually add TE and SA module to the ResNet-50. The results in Table 3 show that both of the module improve the AUC performance on three datasets. As for WDR, CAT achieves the best performance on COIN-SV and Diving48-SV but it is inferior to the vanilla on CSV.

We also visualize the 128-d embedding vectors extracted by different models via PCA in Figure 4. Concretely, we select the first three procedures in the first task of the CSV dataset. Since they only differ in the order of steps but hold the same step set, we can evaluate the effectiveness of the sequence alignment module for handling the order consistency. As shown, the embeddings extracted by the entire CAT model in sub-figure (c) have the largest inter-procedure variance and the smallest averaged intra-procedure variance.

5.4. Performance on Different Splits

As a reminder, the step-level transformation contains deletions, additions, and order exchange of steps. To verify the order-sensitivity of the sequence alignment module, we further re-divide the testing set of CSV into two splits of which one consists of video pairs containing step additions and deletions, and the other consists of pairs containing order exchange of steps, which refers to *alter-number split* and *alter-order split*, respectively.

The results shown in Table 4 indicate that CAT without SA module achieves inferior performance on both *alter-*

Test Split	CAT w/o SA	CAT w/ SA
alter-number	73.01	75.82 (+2.81)
alter-order	80.24	86.32 (+6.08)

Table 4. Results on different test splits. (metric: AUC (%))

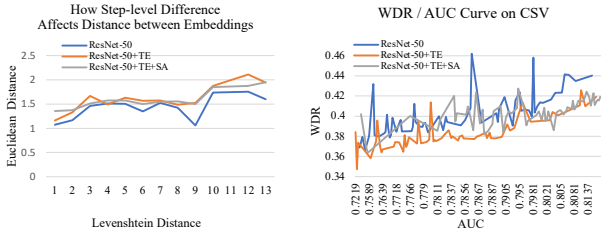


Figure 5. **Left:** Euclidian Distance between procedures with different degrees of step-level divergence in the embedding space; **Right:** Curve of WDR versus AUC on the testing split of CSV.

number split and *alter-order split* while introducing the SA module brings more performance gain on *alter-order split*, which strongly supports the motivation of our proposed SA module that enables the model to be more order-sensitive.

5.5. WDR Curve

To carefully explore the character of our proposed evaluation metric WDR, we conduct two experiments to study the relationship between embedding distance and Levenshtein distance, and the relationship between WDR and AUC. First of all, the curve in Figure 5 **Left** describes a fact that the embedding distance between two procedures increases when the step-level difference denoted by Levenshtein distance gets larger. It is not surprising because the step order is preserved to some extent by all the methods, and large modifications in procedures will lead to distinct embedding differences. Thus, negative pairs with large step-level transformations will dominate the evaluation, which is unfair to those with small ones. To remedy this issue, WDR is introduced and it aims to evaluate embedding distance with respect to unit step-level transformation. Additionally, the curve in Figure 5 **Right** proves the positive correlation between WDR and AUC. The proposed WDR is then expected to be a complementary measurement metric in this new task.

5.6. Scoring Demo

As a potential solution to action assessment, sequence verification is able to act as a judge to score two procedures with fine-grained divergence. Here we show a diving scoring demo in Figure 6. The procedure in V_0 is chosen as the standard reference. We then calculate the cosine similarity as the score between the standard and each candidate video. One can easily tell that V_1 achieves the highest score since it performs the same procedure as V_0, while the scores of V_2 and V_3 decrease with the enlargement of their step-level

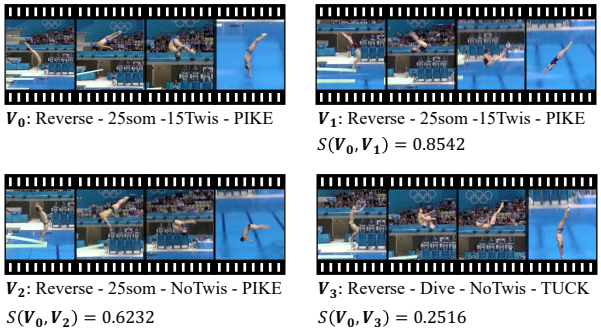


Figure 6. Videos with procedure annotation to be scored. $S(\cdot)$ is the scoring function, which calculates the cosine similarity between two videos in the embedding space.

difference compared to the standard. More demos are available in the supplementary material.

5.7. Limitation and Impact

Though we have introduced two reorganized datasets and collected one scripted dataset for this new task, it still suffers from data insufficiency leading to an unsatisfactory performance on Video Swin Transformer. Also, it may prevent this promising task from being applied to the real application considering the generalization ability in the wild. The transformer encoder is introduced to aggregate temporal information, but it brings a parameter explosion as shown in Table 2. Except for these, we hope this promising task could provide a novel insight for video understanding.

6. Conclusion

In this work, we advocate a novel and interesting task sequence verification developed to verify two procedures with step-level differences when performing the same task. To that end, we reorganize two publicly available action-related datasets with step-procedure-task structure and collect an egocentric dataset asking volunteers to perform various scripted procedures. In addition to that, we develop a new evaluation metric that has been well verified to be a complement to the existing AUC metric. Finally, our proposed transformer-based method has been wildly studied and acts as a strong baseline for this new task.

7. Acknowledgements

The work was supported by National Key R&D Program of China (2018AAA0100704), NSFC #61932020, #62172279, Science and Technology Commission of Shanghai Municipality (Grant No.20ZR1436000), and 'Shuguang Program' supported by Shanghai Education Development Foundation and Shanghai Municipal Education Commission.

References

[1] Sami Abu-El-Haija, Nisarg Kothari, Joonseok Lee, Paul Natsev, George Toderici, Balakrishnan Varadarajan, and Sudheendra Vijayanarasimhan. Youtube-8m: A large-scale video classification benchmark. *arXiv preprint arXiv:1609.08675*, 2016. 3

[2] Jean-Baptiste Alayrac, Piotr Bojanowski, Nishant Agrawal, Josef Sivic, Ivan Laptev, and Simon Lacoste-Julien. Unsupervised learning from narrated instruction videos. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4575–4583, 2016. 3

[3] Humam Alwassel, Silvio Giancola, and Bernard Ghanem. Tsp: Temporally-sensitive pretraining of video encoders for localization tasks. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 3173–3183, 2021. 2

[4] Brandon Amos, Bartosz Ludwiczuk, Mahadev Satyanarayanan, et al. Openface: A general-purpose face recognition library with mobile applications. *CMU School of Computer Science*, 6(2), 2016. 4

[5] Anurag Bagchi, Jazib Mahmood, Dolton Fernandes, and Ravi Kiran Sarvadevabhatla. Hear me out: Fusional approaches for audio augmented temporal action localization. *arXiv preprint arXiv:2106.14118*, 2021. 2

[6] Sara Beery, Guanhang Wu, Vivek Rathod, Ronny Votel, and Jonathan Huang. Context r-cnn: Long term temporal context for per-camera object detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13075–13085, 2020. 1

[7] Piotr Bojanowski, Rémi Lajugie, Francis Bach, Ivan Laptev, Jean Ponce, Cordelia Schmid, and Josef Sivic. Weakly supervised action labeling in videos under ordering constraints. In *European Conference on Computer Vision*, pages 628–643. Springer, 2014. 2, 3

[8] 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. 2, 3

[9] Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. End-to-end object detection with transformers. *arXiv preprint arXiv:2005.12872*, 2020. 5

[10] 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. 2, 3, 4

[11] Yu-Wei Chao, Sudheendra Vijayanarasimhan, Bryan Seybold, David A Ross, Jia Deng, and Rahul Sukthankar. Rethinking the faster r-cnn architecture for temporal action localization. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1130–1139, 2018. 2

[12] RPW Christoph and Feichtenhofer Axel Pinz. Spatiotemporal residual networks for video action recognition. *Advances in Neural Information Processing Systems*, pages 3468–3476, 2016. 3, 4

[13] Rui Dai, Srijan Das, Luca Minciullo, Lorenzo Garattoni, Gianpiero Francesca, and Francois Bremond. Pdan: Pyramid dilated attention network for action detection. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 2970–2979, 2021. 2, 3

[14] Dima Damen, Hazel Doughty, Giovanni Farinella, Sanja Fidler, Antonino Furnari, Evangelos Kazakos, Davide Moltisanti, Jonathan Munro, Toby Perrett, Will Price, et al. The epic-kitchens dataset: Collection, challenges and baselines. *IEEE Transactions on Pattern Analysis & Machine Intelligence*, (01):1–1, 2020. 1

[15] Dima Damen, Hazel Doughty, Giovanni Maria Farinella, Sanja Fidler, Antonino Furnari, Evangelos Kazakos, Davide Moltisanti, Jonathan Munro, Toby Perrett, Will Price, et al. Scaling egocentric vision: The epic-kitchens dataset. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 720–736, 2018. 2, 3

[16] Jiankang Deng, Jia Guo, Yuxiang Zhou, Jinke Yu, Irene Kotsia, and Stefanos Zafeiriou. Retinaface: Single-stage dense face localisation in the wild. *arXiv preprint arXiv:1905.00641*, 2019. 4

[17] Ali Diba, Mohsen Fayyaz, Vivek Sharma, Amir Hossein Karami, Mohammad Mahdi Arzani, Rahman Yousefzadeh, and Luc Van Gool. Temporal 3d convnets: New architecture and transfer learning for video classification. *arXiv preprint arXiv:1711.08200*, 2017. 3, 4

[18] Li Ding and Chenliang Xu. Weakly-supervised action segmentation with iterative soft boundary assignment. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6508–6516, 2018. 3

[19] Jeffrey Donahue, Lisa Anne Hendricks, Sergio Guadarrama, Marcus Rohrbach, Subhashini Venugopalan, Kate Saenko, and Trevor Darrell. Long-term recurrent convolutional networks for visual recognition and description. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2625–2634, 2015. 3, 4

[20] 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. 2, 5

[21] Yong Du, Wei Wang, and Liang Wang. Hierarchical recurrent neural network for skeleton based action recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1110–1118, 2015. 3

[22] Ehsan Elhamifar and Zwe Naing. Unsupervised procedure learning via joint dynamic summarization. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 6341–6350, 2019. 3

[23] Yazan Abu Farha and Jurgen Gall. Ms-tcn: Multi-stage temporal convolutional network for action segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3575–3584, 2019. 3

[24] Mohsen Fayyaz and Jurgen Gall. Sct: Set constrained temporal transformer for set supervised action segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 501–510, 2020. 2

[25] Christoph Feichtenhofer, Haoqi Fan, Jitendra Malik, and Kaiming He. Slowfast networks for video recognition. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 6202–6211, 2019. 2, 3

[26] Christoph Feichtenhofer, Axel Pinz, and Andrew Zisserman. Convolutional two-stream network fusion for video action recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1933–1941, 2016. 3, 4

[27] Daniel Fried, Jean-Baptiste Alayrac, Phil Blunsom, Chris Dyer, Stephen Clark, and Aida Nematzadeh. Learning to segment actions from observation and narration. *arXiv preprint arXiv:2005.03684*, 2020. 3

[28] Jiyang Gao, Zhenheng Yang, and Ram Nevatia. Cascaded boundary regression for temporal action detection. *arXiv preprint arXiv:1705.01180*, 2017. 2

[29] Reza Ghoddoosian, Saif Sayed, and Vassilis Athitsos. Hierarchical modeling for task recognition and action segmentation in weakly-labeled instructional videos. *arXiv preprint arXiv:2110.05697*, 2021. 3

[30] Rohit Girdhar, Joao Carreira, Carl Doersch, and Andrew Zisserman. A better baseline for ava. *arXiv preprint arXiv:1807.10066*, 2018. 2

[31] Raghav Goyal, Samira Ebrahimi Kahou, Vincent Michalski, Joanna Materzynska, Susanne Westphal, Heuna Kim, Valentin Haenel, Ingo Fruend, Peter Yianilos, Moritz Mueller-Freitag, et al. The" something something" video database for learning and evaluating visual common sense. In *ICCV*, volume 1, page 5, 2017. 3

[32] Chunhui Gu, Chen Sun, David A Ross, Carl Vondrick, Caroline Pantofaru, Yeqing Li, Sudheendra Vijayanarasimhan, George Toderici, Susanna Ricco, Rahul Sukthankar, et al. Ava: A video dataset of spatio-temporally localized atomic visual actions. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6047–6056, 2018. 1, 2

[33] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Delving deep into rectifiers: Surpassing human-level performance on imagenet classification. In *Proceedings of the IEEE international conference on computer vision*, pages 1026–1034, 2015. 6

[34] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016. 5

[35] De-An Huang, Shyamal Buch, Lucio Dery, Animesh Garg, Li Fei-Fei, and Juan Carlos Niebles. Finding" it": Weakly-supervised reference-aware visual grounding in instructional videos. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5948–5957, 2018. 3

[36] De-An Huang, Li Fei-Fei, and Juan Carlos Niebles. Connectionist temporal modeling for weakly supervised action labeling. In *European Conference on Computer Vision*, pages 137–153. Springer, 2016. 3

[37] Gabriel Huang, Bo Pang, Zhenhai Zhu, Clara Rivera, and Radu Soricut. Multimodal pretraining for dense video captioning. *arXiv preprint arXiv:2011.11760*, 2020. 3

[38] Haroon Idrees, Amir R Zamir, Yu-Gang Jiang, Alex Gorban, Ivan Laptev, Rahul Sukthankar, and Mubarak Shah. The thumos challenge on action recognition for videos "in the wild". *Computer Vision and Image Understanding*, 155:1–23, 2017. 2

[39] Kumara Kahatapitiya and Michael S Ryoo. Coarse-fine networks for temporal activity detection in videos. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8385–8394, 2021. 2, 3

[40] Andrej Karpathy, George Toderici, Sanketh Shetty, Thomas Leung, Rahul Sukthankar, and Li Fei-Fei. Large-scale video classification with convolutional neural networks. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, pages 1725–1732, 2014. 3

[41] 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. 3, 6

[42] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014. 6

[43] H. Kuehne, A. B. Arslan, and T. Serre. The language of actions: Recovering the syntax and semantics of goal-directed human activities. In *Proceedings of Computer Vision and Pattern Recognition Conference (CVPR), year =*. 2

[44] Hildegard Kuehne, Hueihan Jhuang, Estíbaliz Garrote, Tomaso Poggio, and Thomas Serre. Hmdb: a large video database for human motion recognition. In *2011 International Conference on Computer Vision*, pages 2556–2563. IEEE, 2011. 3

[45] Hilde Kuehne, Alexander Richard, and Juergen Gall. Weakly supervised learning of actions from transcripts. *Computer Vision and Image Understanding*, 163:78–89, 2017. 3

[46] Anna Kukleva, Hilde Kuehne, Fadime Sener, and Jurgen Gall. Unsupervised learning of action classes with continuous temporal embedding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12066–12074, 2019. 2

[47] Colin Lea, Michael D Flynn, Rene Vidal, Austin Reiter, and Gregory D Hager. Temporal convolutional networks for action segmentation and detection. In *proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 156–165, 2017. 2

[48] Colin Lea, Austin Reiter, René Vidal, and Gregory D Hager. Segmental spatiotemporal cnns for fine-grained action segmentation. In *European Conference on Computer Vision*, pages 36–52. Springer, 2016. 3

[49] Wei Li, Zehuan Yuan, Dashan Guo, Lei Huang, Xiangzhong Fang, and Changhu Wang. Deformable tube network for action detection in videos. *arXiv preprint arXiv:1907.01847*, 2019. 2

[50] Xin Li, Tianwei Lin, Xiao Liu, Wangmeng Zuo, Chao Li, Xiang Long, Dongliang He, Fu Li, Shilei Wen, and Chuang Gan. Deep concept-wise temporal convolutional networks for action localization. In *Proceedings of the 28th ACM In-*

ternational Conference on Multimedia, pages 4004–4012, 2020. 2

[51] Yingwei Li, Yi Li, and Nuno Vasconcelos. Resound: Towards action recognition without representation bias. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 513–528, 2018. 2, 3, 4

[52] Ji Lin, Chuang Gan, and Song Han. Tsm: Temporal shift module for efficient video understanding. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 7083–7093, 2019. 2, 3, 6, 7

[53] Tianwei Lin, Xiao Liu, Xin Li, Errui Ding, and Shilei Wen. Bmn: Boundary-matching network for temporal action proposal generation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 3889–3898, 2019. 2, 3

[54] Tianwei Lin, Xu Zhao, Haisheng Su, Chongjing Wang, and Ming Yang. Bsn: Boundary sensitive network for temporal action proposal generation. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 3–19, 2018. 2, 3

[55] Ze Liu, Jia Ning, Yue Cao, Yixuan Wei, Zheng Zhang, Stephen Lin, and Han Hu. Video swin transformer. *arXiv preprint arXiv:2106.13230*, 2021. 1, 6, 7

[56] Huaishao Luo, Lei Ji, Botian Shi, Haoyang Huang, Nan Duan, Tianrui Li, Jason Li, Taroon Bharti, and Ming Zhou. Univl: A unified video and language pre-training model for multimodal understanding and generation. *arXiv preprint arXiv:2002.06353*, 2020. 3

[57] Effrosyni Mavroudi, Benjamín Béjar Haro, and René Vidal. Representation learning on visual-symbolic graphs for video understanding. In *European Conference on Computer Vision*, pages 71–90. Springer, 2020. 2, 3

[58] Antoine Miech, Jean-Baptiste Alayrac, Lucas Smaira, Ivan Laptev, Josef Sivic, and Andrew Zisserman. End-to-end learning of visual representations from uncurated instructional videos. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9879–9889, 2020. 3

[59] Mathew Monfort, Alex Andonian, Bolei Zhou, Kandan Ramakrishnan, Sarah Adel Bargal, Tom Yan, Lisa Brown, Quanfu Fan, Dan Gutfreund, Carl Vondrick, et al. Moments in time dataset: one million videos for event understanding. *IEEE transactions on pattern analysis and machine intelligence*, 42(2):502–508, 2019. 2

[60] Naila Murray, Luca Marchesotti, and Florent Perronnin. Ava: A large-scale database for aesthetic visual analysis. In *2012 IEEE Conference on Computer Vision and Pattern Recognition*, pages 2408–2415. IEEE, 2012. 3

[61] AJ Piergiovanni, Anelia Angelova, Michael S Ryoo, and Irfan Essa. Unsupervised action segmentation for instructional videos. *arXiv preprint arXiv:2106.03738*, 2021. 3

[62] AJ Piergiovanni and Michael Ryoo. Temporal gaussian mixture layer for videos. In *International Conference on Machine learning*, pages 5152–5161. PMLR, 2019. 2, 3

[63] AJ Piergiovanni and Michael S Ryoo. Learning latent super-events to detect multiple activities in videos. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5304–5313, 2018. 2, 3

[64] AJ Piergiovanni and Michael S Ryoo. Avid dataset: Anonymized videos from diverse countries. *arXiv preprint arXiv:2007.05515*, 2020. 2, 3

[65] Zhaofan Qiu, Ting Yao, and Tao Mei. Learning spatio-temporal representation with pseudo-3d residual networks. In *proceedings of the IEEE International Conference on Computer Vision*, pages 5533–5541, 2017. 3, 4

[66] Alexander Richard and Juergen Gall. Temporal action detection using a statistical language model. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3131–3140, 2016. 3

[67] Alexander Richard, Hilde Kuehne, and Juergen Gall. Weakly supervised action learning with rnn based fine-to-coarse modeling. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, pages 754–763, 2017. 3

[68] Marcus Rohrbach, Sikandar Amin, Mykhaylo Andriluka, and Bernt Schiele. A database for fine grained activity detection of cooking activities. In *2012 IEEE Conference on Computer Vision and Pattern Recognition*, pages 1194–1201. IEEE, 2012. 3

[69] Michael S Ryoo, AJ Piergiovanni, Mingxing Tan, and Anelia Angelova. Assemblenet: Searching for multi-stream neural connectivity in video architectures. *arXiv preprint arXiv:1905.13209*, 2019. 1

[70] Florian Schroff, Dmitry Kalenichenko, and James Philbin. Facenet: A unified embedding for face recognition and clustering. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 815–823, 2015. 4

[71] Ozan Sener, Amir R Zamir, Silvio Savarese, and Ashutosh Saxena. Unsupervised semantic parsing of video collections. In *Proceedings of the IEEE International conference on Computer Vision*, pages 4480–4488, 2015. 3

[72] Yuhan Shen, Lu Wang, and Ehsan Elhamifar. Learning to segment actions from visual and language instructions via differentiable weak sequence alignment. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10156–10165, 2021. 2, 3

[73] Qinfeng Shi, Li Wang, Li Cheng, and Alex Smola. Discriminative human action segmentation and recognition using semi-markov model. In *2008 IEEE Conference on Computer Vision and Pattern Recognition*, pages 1–8. IEEE, 2008. 3

[74] Zheng Shou, Jonathan Chan, Alireza Zareian, Kazuyuki Miyazawa, and Shih-Fu Chang. Cdc: Convolutional-de-convolutional networks for precise temporal action localization in untrimmed videos. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5734–5743, 2017. 2

[75] Zheng Shou, Dongang Wang, and Shih-Fu Chang. Temporal action localization in untrimmed videos via multi-stage cnns. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1049–1058, 2016. 2, 3

[76] Gunnar A Sigurdsson, Jean-Baptiste Alayrac, Aida Nematzadeh, Lucas Smaira, Mateusz Malinowski, Joao Carreira, Phil Blunsom, and Andrew Zisserman. Visual

grounding in video for unsupervised word translation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10850–10859, 2020. 3

[77] Gunnar A Sigurdsson, Santosh Divvala, Ali Farhadi, and Abhinav Gupta. Asynchronous temporal fields for action recognition. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, pages 585–594, 2017. 2

[78] Gunnar A Sigurdsson, Gül Varol, Xiaolong Wang, Ali Farhadi, Ivan Laptev, and Abhinav Gupta. Hollywood in homes: Crowdsourcing data collection for activity understanding. In *European Conference on Computer Vision*, pages 510–526. Springer, 2016. 2, 3

[79] Karen Simonyan and Andrew Zisserman. Two-stream convolutional networks for action recognition in videos. In *Advances in neural information processing systems*, pages 568–576, 2014. 3, 4

[80] Sijie Song, Cuiling Lan, Junliang Xing, Wenjun Zeng, and Jiaying Liu. An end-to-end spatio-temporal attention model for human action recognition from skeleton data. *arXiv preprint arXiv:1611.06067*, 2016. 3

[81] Yale Song, Jordi Vallmitjana, Amanda Stent, and Alejandro Jaimes. Tvsum: Summarizing web videos using titles. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5179–5187, 2015. 3

[82] Khurram Soomro, Amir Roshan Zamir, and Mubarak Shah. Ucf101: A dataset of 101 human actions classes from videos in the wild. *arXiv preprint arXiv:1212.0402*, 2012. 2, 3

[83] Chen Sun, Abhinav Shrivastava, Carl Vondrick, Kevin Murphy, Rahul Sukthankar, and Cordelia Schmid. Actor-centric relation network. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 318–334, 2018. 2, 3

[84] Jiajun Tang, Jin Xia, Xinzhi Mu, Bo Pang, and Cewu Lu. Asynchronous interaction aggregation for action detection. In *European Conference on Computer Vision*, pages 71–87. Springer, 2020. 2

[85] Yansong Tang, Dajun Ding, Yongming Rao, Yu Zheng, Danyang Zhang, Lili Zhao, Jiwen Lu, and Jie Zhou. Coin: A large-scale dataset for comprehensive instructional video analysis. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1207–1216, 2019. 1, 2, 3, 4

[86] Zineng Tang, Jie Lei, and Mohit Bansal. Decembert: Learning from noisy instructional videos via dense captions and entropy minimization. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 2415–2426, 2021. 3

[87] 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. 3, 4

[88] Du Tran, Jamie Ray, Zheng Shou, Shih-Fu Chang, and Manohar Paluri. Convnet architecture search for spatiotemporal feature learning. *arXiv preprint arXiv:1708.05038*, 2017. 3, 4

[89] 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. 5

[90] Limin Wang, Yuanjun Xiong, Zhe Wang, Yu Qiao, Dahua Lin, Xiaoou Tang, and Luc Van Gool. Temporal segment networks: Towards good practices for deep action recognition. In *European conference on computer vision*, pages 20–36. Springer, 2016. 2, 3, 4, 5, 6, 7

[91] Xiang Wang, Zhiwu Qing, Ziyuan Huang, Yutong Feng, Shiwei Zhang, Jianwen Jiang, Mingqian Tang, Changxin Gao, and Nong Sang. Proposal relation network for temporal action detection. *arXiv preprint arXiv:2106.11812*, 2021. 2

[92] Junwu Weng, Chaoqun Weng, and Junsong Yuan. Spatio-temporal naive-bayes nearest-neighbor (st-nbnn) for skeleton-based action recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4171–4180, 2017. 3

[93] Bichen Wu, Chenfeng Xu, Xiaoliang Dai, Alvin Wan, Peizhao Zhang, Masayoshi Tomizuka, Kurt Keutzer, and Peter Vajda. Visual transformers: Token-based image representation and processing for computer vision. *arXiv preprint arXiv:2006.03677*, 2020. 5

[94] Chao-Yuan Wu, Christoph Feichtenhofer, Haoqi Fan, Kaiming He, Philipp Krahenbuhl, and Ross Girshick. Long-term feature banks for detailed video understanding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 284–293, 2019. 2

[95] Yuanjun Xiong, Yue Zhao, Limin Wang, Dahua Lin, and Xiaoou Tang. A pursuit of temporal accuracy in general activity detection. *arXiv preprint arXiv:1703.02716*, 2017. 2

[96] Huijuan Xu, Abir Das, and Kate Saenko. R-c3d: Region convolutional 3d network for temporal activity detection. In *Proceedings of the IEEE international conference on computer vision*, pages 5783–5792, 2017. 2, 3

[97] Mengmeng Xu, Chen Zhao, David S Rojas, Ali Thabet, and Bernard Ghanem. G-tad: Sub-graph localization for temporal action detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10156–10165, 2020. 2

[98] Sijie Yan, Yuanjun Xiong, and Dahua Lin. Spatial temporal graph convolutional networks for skeleton-based action recognition. *arXiv preprint arXiv:1801.07455*, 2018. 3

[99] Serena Yeung, Olga Russakovsky, Greg Mori, and Li Fei-Fei. End-to-end learning of action detection from frame glimpses in videos. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2678–2687, 2016. 2, 3

[100] Fangqiu Yi, Hongyu Wen, and Tingting Jiang. Asformer: Transformer for action segmentation. In *The British Machine Vision Conference (BMVC)*, 2021. 2

[101] Runhao Zeng, Wenbing Huang, Mingkui Tan, Yu Rong, Peilin Zhao, Junzhou Huang, and Chuang Gan. Graph convolutional networks for temporal action localization. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 7094–7103, 2019. 2

[102] Chen Zhao, Ali K Thabet, and Bernard Ghanem. Video self-stitching graph network for temporal action localization. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 13658–13667, 2021. 2

[103] Yue Zhao, Yuanjun Xiong, Limin Wang, Zhirong Wu, Xiaoou Tang, and Dahua Lin. Temporal action detection with structured segment networks. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 2914–2923, 2017. 3

[104] Bolei Zhou, Alex Andonian, Aude Oliva, and Antonio Torralba. Temporal relational reasoning in videos. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 803–818, 2018. 2, 3, 6, 7

[105] Luowei Zhou, Chenliang Xu, and Jason J Corso. Towards automatic learning of procedures from web instructional videos. In *Thirty-Second AAAI Conference on Artificial Intelligence*, 2018. 3

[106] Xizhou Zhu, Weijie Su, Lewei Lu, Bin Li, Xiaogang Wang, and Jifeng Dai. Deformable detr: Deformable transformers for end-to-end object detection. *arXiv preprint arXiv:2010.04159*, 2020. 5

[107] Dimitri Zhukov, Jean-Baptiste Alayrac, Ramazan Gokberk Cinbis, David Fouhey, Ivan Laptev, and Josef Sivic. Cross-task weakly supervised learning from instructional videos. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3537–3545, 2019. 2