On Temporal Granularity in Self-Supervised Video Representation Learning

Rui Qian[1,2]
https://rui1996.github.io/
Yeqing Li[1]
yeqing@google.com
Liangzhe Yuan[1]
lzyuan@google.com
Boqing Gong[1]
bgong@google.com
Ting Liu[1]
http://www.tliu.org/
Matthew Brown[1]
mtbr@google.com
Serge Belongie[3]
s.belongie@di.ku.dk
Ming-Hsuan Yang[1]
minghsuan@google.com
Hartwig Adam[1]
hadam@google.com
Yin Cui[1]
https://ycui.me/

[1] Google Research
[2] Cornell University
[3] University of Copenhagen

Abstract

This work presents an empirical exploration of temporal granularity in self-supervised video representation learning. While state-of-the-art methods commonly enforce the learned features to be temporally-persistent across the whole video, we argue that this objective may not be suitable for all video tasks. To reveal the impact of temporal granularity, we propose a simple unified framework to learn features from same unlabeled videos with varying granularities from temporally fine-grained to persistent, by only adjusting one coefficient. We conduct a comprehensive empirical study covering a variety of classic and emerging video benchmarks and find video-level understanding tasks prefer temporally persistent features while temporal understanding inside one video favors fine-grained features. The flexibility of our framework gives rise to competitive or state-of-the-art performance, even outperforming supervised pre-training in a few cases. Code will be available at https://github.com/tensorflow/models/tree/master/official/.

Figure 1: Illustration of tasks requiring different temporal granularities on the same video. Event boundary detection requires temporally fine-grained features, while video-level recognition prefers temporally persistent features.

1 Introduction

Learning visual representations from abundantly available unlabeled videos is of crucial importance in computer vision. Thanks to the breakthrough in image self-supervised learning [9, 13, 24, 28], a series of recent works extended similar ideas to video [18, 48, 49]. The success of these methods largely depends on a seemly counter-intuitive objective: enforcing temporal persistency across an entire video [18, 48, 49].

Despite the strong performance on commonly used video benchmarks (*e.g.*, action recognition [32, 35, 61]), we argue that this temporal persistency objective is not always preferable, especially on tasks that require fine-grained temporal understanding inside a video. Consider an example in Fig. 1. Event boundary detection calls for temporally fine-grained features so that the model is aware of the temporal content shifts within the video. In contrast, video-level event recognition requires the model to robustly predict the target label based on some sampled clips; therefore, temporally persistent/coarse-grained features are more desirable. How can we develop a self-supervised video representation learning framework that accounts for both fine-grained and persistent temporal information?

We try to answer the above question by considering temporal granularity. The concept of temporal granularity has been studied in speech recognition [21] and time series analysis [5, 15], but is rather under-explored in recent video representation learning research. In this paper, we aim at learning a set of features with coarse to fine temporal granularities from the same videos to understand the impact of temporal granularity. To achieve this goal, we propose **TeG**, a framework to explore **Te**mporal **G**ranularity via the combination of fine-grained and persistent temporal learning, as illustrated in Fig. 2.

In TeG, we randomly sample a long clip from a video and a short clip that lies inside the time duration of the long clip. We then feed them into a video encoder without temporal average pooling, maintaining their temporal resolution. The resultant features are projected into two separate embedding spaces with different contrastive learning objectives.

In the fine-grained temporal learning space, we split the projected features along the temporal dimension into a list of temporal embeddings, each represents the feature of a short time duration. We apply a dense contrastive objective to maximize the similarity between corresponding temporal embeddings from two clips, making the learned features to be temporally discriminative within a clip.

In the persistent temporal learning space, we directly apply a global average pooling to generate the global embedding for both the short clip and the long clip. The training objective here encourages global temporal persistency by pulling together two embeddings, similarly to what has been used in existing frameworks [18, 48, 49].

TeG optimizes both fine-grained and persistent temporal learning objectives and offers a flexible solution to learning features of different temporal granularities by adjusting the loss weight coefficient between the two objectives.

We conduct comprehensive experiments on commonly used video benchmarks together with two emerging benchmarks for understanding events in short videos: VidSitu event classification [52] and Kinetics-GEBD (generic event boundary detection) [56]. We find that tasks that require fine-grained temporal understanding inside one video like VidSitu event classification and Kinetics-GEBD prefer temporally fine-grained features. Bringing in temporally persistent features hurt the performance, see Tab. 1. On the contrary, tasks of video-level classification are generally in favour of temporally persistent features, see Tab. 2 and Tab. 3. Features learned from our unified framework achieve very competitive performance: 67.8% on Kinetics-400 linear evaluation, 94.1% on UCF101, 71.9% on HMDB51, 71.4% F-1 score on Kinetics-GEBD, 28.7% mAP on AVA-Kinetics.

2 Related Work

Unsupervised video representation learning. In an early work, Srivastava *et al.* [62] propose to predict the future based on frame features. More recent works learn from raw videos by predicting motion and appearance statistics [66], speed [7, 67] and encodings [25, 27, 43]. Aside from future prediction, it is common to learn from pretext tasks like sorting frames or video clips [20, 33, 37, 70] and rotation [31]. Recently, constrastive learning based methods [6, 18, 39, 48, 49, 60, 65, 68] significantly reduce the gap with supervised learning by pulling together features of clips from the same video. Furthermore, videos containing multimodal signals make it possible to learn from speech or language [44, 63, 64], audio [3, 4, 34, 46], optical flow [26], or combinations of modalities [1, 2, 49] and tasks [47]. Different from existing work, we introduce temporally fine-grained features into the video contrastive learning framework and study its impact on various downstream tasks.

Fine-grained temporal video understanding. We first discuss two representative tasks: temporal localization and segmentation. Commonly used temporal localization benchmarks (*e.g.*, ActivityNet [8], THUMOS [30], HACS [73]) are constructed based on specified action classes. As a result, most temporal localization methods [40, 41, 42, 57, 58, 74] contain a temporal proposal module to simply treat video segments that do not belong to pre-defined classes as the background. Temporal segmentation methods [17, 36, 51] typically divide a video into segments of actions, or sub-actions [53, 54]. But still, those methods can only predict boundaries of pre-defined classes, not generic boundaries. We choose the recently proposed Kinetics-GEBD [56] dataset to verify whether TeG is able to learn temporally fine-grained features that can be used for generic event boundary detection. We also benchmark our method on AVA-Kinetics [38] for spatiotemporal action localization. In addition, movies could also provide rich content for fine-grained temporal video understanding. However, temporal movie understanding methods [12, 29, 45] typically focus on shots (sharp transitions due to video editing) and can be accurately localized using low-level visual cues [59]. To benchmark TeG in movie scenes, we adopt the recently proposed VidSitu [52] dataset, in which each short video is temporally annotated with 5 events with natural transitions.

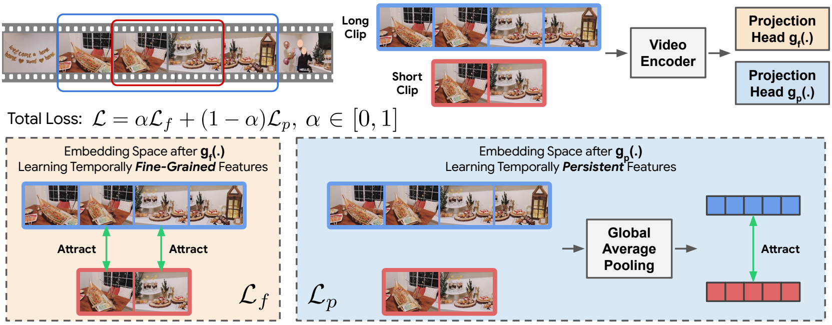


Figure 2: Overview of TeG framework. We randomly sample a long clip from a video and a short clip that lies inside the duration of the long clip. We then project encoded features into two separate embedding spaces, one for learning temporally fine-grained features and the other for temporally persistent features.

3 Method

An overview of our framework is shown in Fig. 2. We next introduce each component.

Temporal sampling. Given a video of N frames, $V = \{v_1, v_2, \cdots, v_N\}$, we adopt a long-short sampling strategy, where we first sample a long clip l randomly from the whole video, and then a short clip s inside the time duration of the long clip. The long clip provides rich spatiotemporal context, and the short clip in it guarantees that each temporal embedding in the short clip has a corresponding temporal embedding in the long clip at approximately the same start and end time. The ablation on sampling strategy is in Tab. 5(a).

Spatial data augmentation. After obtaining the short clip s and long clip l, we adopt the common practice in recent video contrastive learning [2, 3, 48] of applying spatial data augmentations including random resizing and cropping, color jittering, and Gaussian blurring.

Video encoder. We adopt the 3D-ResNet-50 (R3D-50) backbone used in [48] and remove the final *temporal* average pooling to maintain the temporal resolution of features. We apply two projection heads: $g_p(\cdot)$ for persistent temporal learning and $g_f(\cdot)$ for fine-grained temporal learning. They project representations into separate embedding spaces with different contrastive objectives. In the persistent learning space, we obtain embedding z_p^s from the short clip s and z_p^l from the long clip l by $\{z_p^s, z_p^l\} = \{g_p(f(s)), g_p(f(l))\}$; in the fine-grained learning space, we have $\{z_f^s, z_f^l\} = \{g_f(f(s)), g_f(f(l))\}$.

Our approach maintains a simple form of video contrastive learning where we do not use separate encoders for different clips [49], nor do we use a momentum encoder [18], predictor head [18, 68] and symmetric losses [49]. Extensive experiments in Sec. 5 demonstrate the effectiveness of this simple design.

Temporal aggregation. For temporally persistent learning, as a common practice [18, 48], we directly apply a global average pooling to get a single vector representing the whole clip, resulting in $z_p^s, z_p^l \in \mathbb{R}^{1\times c}$, where c is the number of output channels from the projection head. For temporally fine-grained learning, we design a configurable local aggregation strategy to optionally aggregate consecutive local temporal embeddings to reduce training complexity. We denote the number of frames in short clip s and long clip l as T_s and T_l. The aggregation performs average pooling on every consecutive $\frac{T_s}{n}$ frames in the short clip and $\frac{T_l}{m}$ frames in the long clip, resulting in aggregated outputs of $z_f^s \in \mathbb{R}^{n\times c}$ and $z_f^l \in \mathbb{R}^{m\times c}$. When $n = 1$ and $m = 1$, it reduces to temporal persistent learning. When $n = T_S$ and $m = T_L$, it conducts dense temporal contrastive learning on frame-level embeddings. Fig. 5(b) ablates on the different choices of n and m. We use $z_f^s[i]$ to index the i-th dimension of z_f^s and $z_f^l[j]$ to index the j-th dimension of z_f^l, where $1 \leq i \leq n$ and $1 \leq j \leq n$.

Fine-grained temporal learning. We aim to obtain temporally fine-grained features by maximizing the feature similarity between corresponding embeddings of the short and the long clip. The corresponding embeddings should be close in time and we rely on the frame index to find them. After temporal aggregation on a few consecutive frames, we define the index of a certain embedding $z_f^s[i]$ as the average frame index of all aggregated frames, notated as $I(z_f^s[i])$. We find $z_f^s[i]$'s nearest corresponding embedding $z_f^l[j]$ in the long clip by: $j = \arg\min_j |I(z_f^s[i]) - I(z_f^l[j])|$. $(z_f^s[i], z_f^l[j])$ has the closest temporal distance and it is considered as the positive pair. The fine-grained temporal learning loss can be written as:

$$\mathcal{L}_f = -\frac{1}{n}\sum_{i=1}^{n} \log \frac{\exp(z_f^s[i] \cdot z_f^l[j]/\tau)}{\exp(z_f^s[i] \cdot z_f^l[j]/\tau) + \sum_{k_f^-} \exp(z_f^s[i] \cdot k_f^-/\tau))}, \tag{1}$$

where k_f^- represents all dense embeddings of long clips from other videos after temporal aggregation in the fine-grained temporal learning space and τ is the temperature.

Persistent temporal learning. Recall that we have embeddings $z_p^s, z_p^l \in \mathbb{R}^{1\times c}$ in the temporally persistent learning space. (z_p^s, z_p^l) is considered as the positive pair and (z_p^s, k_p^-) as negative pairs, where k_p^- represents all global embeddings from long clips of other videos in the embedding space. The persistent temporal learning loss can be written as:

$$\mathcal{L}_p = -\log \frac{\exp(z_p^s \cdot z_p^l/\tau)}{\exp(z_p^s \cdot z_p^l/\tau) + \sum_{k_p^-} \exp(z_p^s \cdot k_p^-/\tau)}. \tag{2}$$

For simplicity, we use the same temperature τ for both $\mathcal{L}_f$ and $\mathcal{L}_p$.

Total loss. The total loss is a weighted sum of fine-grained and persistent learning loss:

$$\mathcal{L} = \alpha \mathcal{L}_f + (1-\alpha)\mathcal{L}_p, \tag{3}$$

where the weight coefficient $\alpha \in [0,1]$ is used to control the temporal granularity of the learned features. When α is close to 0, we intend to learn temporally persistent features with only $\mathcal{L}_p$ in the loss. With the increasing of α, we obtain more temporally fine-grained features. An ablation regarding the effect of α on two datasets is presented in Fig. 4.

4 Evaluation

We describe how we evaluate our method on two new datasets VidSitu [52] and Kinetics-GEBD [56]. See Sec. 5 for the evaluation on other 4 commonly used datasets, including Kinetics via linear probing and various downstream tasks via fine-tuning.

Event classification. VidSitu [52] focuses on understanding the relationship of events in movie videos. Each video in VidSitu is 10-second long and is divided into 5 consecutive non-overlapping events. Each event is annotated with a verb to describe the most salient action. The baseline provided by the original authors is to first cut the video into 5 events according to the annotated boundaries and then perform classification for each event. In

method		backbone	acc.
Supervised	Train from scratch [52]	I3D	31.2
		I3D + NL	30.2
		R3D-50 + NL	33.1
		SlowFast + NL	32.6
Unsupervised	CVRL [48]	R3D-50	28.3
	TeG-PS	R3D-50	28.3
	TeG-FG	R3D-50	**31.1**

(a) **Event classification on Vidsitu**

method	external data	finetuning	F1 score
SceneDet [11]	-	✗	27.5
BMN-SE [41]	IN + THUMOS	✗	49.1
TCN [36]	IN	✗	58.8
PC [56]	IN	✓	62.5
CVRL [48]	-	✓	69.1
TeG-PS	-	✓	69.9
TeG-FG	-	✓	**71.4**

(b) **Event boundary detection on GEBD**

Table 1: In (a) event classification on Vidsitu, NL indicates for non-local block [69]. In (b) event boundary detection on Kinetics-GEBD, IN represents ImageNet supervised pre-training and THUMOS means additional supervised training on THUMOS [30]. TeG-FG with fine-grained temporal learning shows superior performance.

our case, we directly apply our method on raw videos in VidSitu without using any labels in pre-training. Since the transition between events is usually natural and continuous, we consider VidSitu a good benchmark to evaluate whether our method can learn more fine-grained temporal features than video-level persistent learning methods (*e.g.*, [48]). We adopt linear probing during the evaluation, where we use their event labels to train a linear classifier on top the frozen backbone to quantify the performance of the learned representations.

Generic event boundary detection. Kinetics-GEBD [56] annotates Kinetics-400 videos with fine-grained event boundaries based on human perception. Each video receives around five annotated temporal boundaries. A detection is considered correct when its temporal distance with a ground truth is less than 5% of the total video length. We use a 1D sliding window detection method, following the spirit of classic object detection methods like DPM [19]. We first pre-train our backbone without using any annotations. We then add a binary classifier on top of the pre-trained backbone to predict whether a clip contains a boundary or not. Similar to object detection [23, 50], we fine-tune the model end-to-end to benchmark the performance of our learned features.

5 Experiments

As we have introduced that our framework is flexible at learning features with varying granularities, we mainly adopt two representative settings: 1) $\alpha = 0.0$ for persistent temporal learning only and we call this method **TeG-PS**, where PS represents "persistent". 2) $\alpha = 0.9$, in which the fine-grained temporal learning loss is the dominant loss and we denote this method as **TeG-FG**, where FG represents "fine-grained". We also provide an in-depth study for more different values of α on VidSitu and Kinetics in Fig. 4.

5.1 Event Classification

We conduct experiments on VidSitu [52], which contains 23.6k training and 1.3k validation videos with 1560 verb classes. During pre-training, we sample a 32-frame long clip with a stride of 4 and a 16-frame short clip with a stride of 2. Temporal aggregation parameters are set as $m = 4$ and $n = 1$ (ablation study in Fig. 5(b)). We pre-train our model from scratch for 200 epochs on unlabeled raw videos. During the linear evaluation, we train a linear classifier with an initial learning rate of 4.0 for 100 epochs. Additional details can be found in Appendix B.1.

We show TeG's performance on VidSitu in Tab. 1(a). The supervised methods directly train models from scratch on the training set, using labels for each event clip cut from raw videos. The unsupervised methods perform pre-training on raw videos from scratch without using any labels and then conduct linear evaluation. We adopt CVRL [48] as an important baseline since it is a representative method that enforces temporal persistency across the whole video. Despite different settings, TeG-PS actually achieves identical performance with CVRL, indicates the performance of temporally persistent features can be quite similar on VidSitu. By contrast, TeG-FG equipped with temporally fine-grained pre-training improves the performance by 2.8%. Furthermore, the performance TeG-FG is on par with supervised methods using I3D as the backbone. This result provides a solid evidence that temporal persistent learning is not the optimal solution on this event classification benchmark.

We also provide a visualization of feature similarity in Fig. 3. For each event inside the same video, we sample a clip in the middle and feed it into the trained video encoder to get the feature vector. We then calculate the cosine similarities between all pairs of features. For Fig. 3(a), each event has a different label and we observe the fine-grained features are much more discriminative with siginificantly lower similarity scores between different events. As in Fig. 3(b), both features show similar scores within the same label (smoke and talk), while the fine-grained features are more discriminative between the two labels. We provide more visualization examples in Appendix C.

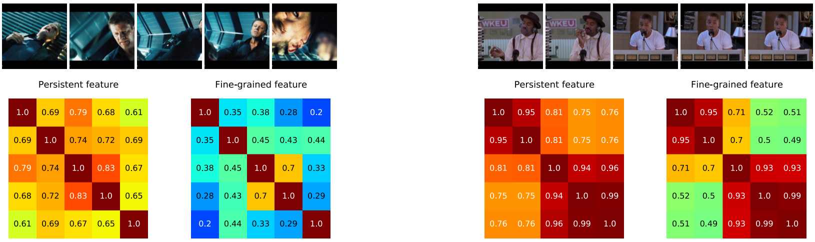
5.2 Generic Event Boundary Detection

We perform experiments on Kinetics-GEBD [56], which contains 20k out of 240k Kinetics-400 [32] training videos and all 20k validation videos. We sample a 16-frame long clip and a 8-frame short clip. We pre-train our model from scratch for 200 epochs and then fine-tune the model with the annotated boundaries for 30 epochs. Other training and evaluation hyper parameters follow the setting of original authors [56] and we would cover more details in Appendix B.2.

TeG's performance on Kinetics-GEBD is presented in Tab. 1(b), where we report results using their strictest temporal threshold of 0.05 to emphasize on the importance of precise boundary detection. We next briefly introduce a few representative methods for this benchmark. SceneDet [11] is a widely-used library for detecting shot changes. BMN-SE [41] is a state-of-the-art method for action proposal generation and here the start and end of each proposal are considered as event boundaries. TCN [36] is a classic action boundary detection method. PC [56] is the state-of-the-art method on this benchmark provided by performing pairwise classification around event boundaries. We group these methods by the external data they pre-train on and whether they fine-tune or keep the backbone frozen and fine-tuning methods achieve much better performances. Compared with PC which relies on ImageNet supervised pre-training, CVRL and TeG can directly pre-train on the training videos without using labels and external data for supervision. We draw a similar observation with event classification that TeG-FG with temporally fine-grained learning outperforms methods enforcing temporal persistency like CVRL and TeG-PS.

5.3 Kinetics Linear Evaluation

We pre-train our model from scratch for 800 epochs on Kinetics-400 [32] with the same parameters with event classification. We perform linear evaluation to directly quantify the learned feature quality, following [18, 48]. As shown in Tab. 2, TeG-FG obtains 65.0% top-1 accuracy which trails behind some state-of-the-art methods including CVRL [48] and



(a) **yell**, **watch**, **point**, **kneel**, **writhe**

(b) **smoke**, **smoke**, **talk**, **talk**, **talk**

Figure 3: Visualization of feature similarity. Top row shows the center frame of input clip. The left matrix is the similarity of temporally persistent features and the right one comes from temporally fine-grained features. Ground truth labels are in subcaptions. Video (a) has different labels for each event, while (b) only has two distinct labels.

method	backbone	pre-train				acc.
		dataset	epoch	frame	FLOPs (G)	
SimCLR [48]	R3D-50	K400	-	-	-	46.8
ImageNet [48]	R3D-50	IN	-	-	-	53.5
SeCo [72]	R-50	IN† + K400	400	-	-	61.9
CVRL [48]	R3D-50$\downarrow$	K400	800	16+16 $\rightarrow$ 8+8	91.2	66.1
ρBYOL (ρ=2) [18]	R3D-50	K400	800	8+8	83.5	66.2
ρMoCo (ρ=2) [18]	R3D-50	K400	800	8+8	83.5	67.4
ρMoCo (ρ=2) [18]	R3D-50	K400	200	16+16	167.0	67.6
ρBYOL (ρ=4) [18]	R3D-50	K400	200	16+16+16+16	**334.0**	**71.4**
TeG-FG	R3D-50$\downarrow$	K400	800	16+32 $\rightarrow$ 8+16	136.4	65.0
TeG-PS	R3D-50$\downarrow$	K400	800	16+32 $\rightarrow$ 8+16	136.4	**67.8**

Table 2: **Linear evaluation on Kinetics-400 action recognition.** We list the number of frames and FLOPs in pre-training stage. We report total FLOPs considering all clips instead of just one clip. R3D-50$\downarrow$ means the first layer conducts an additional 2$\times$ temporal downsampling to approximately reduce the computation to half (shown in frame). IN† denotes a MoCo-v2 [14] checkpoint pre-trained on ImageNet is used as backbone initialization.

ρMoCo [18]. TeG-PS achieves 67.8%, an improvement of 2.8% over TeG-FG. This is a competitive performance on Kinetics linear evaluation without using multi-clip sampling [18] in pre-training. These results verify that temporal persistency in the key to obtain strong performance on Kinetics. Our primary goal here is to investigate the difference between temporally persistent and fine-grained features with strong baselines, not solely competing for best numbers. We believe the performance of TeG-PS could be additionally boosted by multi-clip sampling (*e.g.* $\rho = 4$ of [18]) as it further enhances temporal persistency. Different from [18] reporting only one clip, we report the total FLOPs for all clips used during pre-training in Tab. 2,. Our R3D-50 also has an additional 2$\times$ temporal downsampling in the first layer, reducing the frames to 8+16 in the backbone after the first layer. We offer a more detailed discussion in Appendix A.

5.4 Downstream Action Recognition and Localization

For downstream action recognition, we fine-tune the same pre-trained checkpoint used in Kinetics linear evaluation on UCF101 [61] and HMDB51 [35], which are classic benchmarks for evaluating self-supervised video representation learning.

We report TeG's performance on in Tab. 3(a, b). On UCF, TeG-PS achieves a competitive performance of 94.1% with fine-tuning and 91.1% with linear evaluation. On HMDB, TeG-PS achives 71.9% with fine-tuning and 64.2% with linear evaluation, surpassing CVRL[48] by 4.0% and 5.9%, respectively. TeG-FG does not help on these datasets.

method	pre-train data	UCF	HMDB
MotionPred [66]	K400	61.2	33.4
3D-RotNet [31]	K400	64.5	34.3
ST-Puzzle [33]	K400	65.8	33.7
ClipOrder [70]	K400	72.4	30.9
DPC [25]	K400	75.7	35.7
PacePred [67]	K400	77.1	36.6
MemDPC [27]	K400	78.1	41.2
SpeedNet [7]	K400	81.1	48.8
CoCLR [26]	K400	87.9	54.6
DynamoNet [16]	YT8M	88.1	59.9
SeCo [72]	IN† + K400	88.3	55.6
CVRL [48]	K400	92.9	67.9
MCL [39]	IN† + K400	93.4	69.1
ρMoCo (ρ=4) [18]	K400	93.6	-
TeG-FG	K400	93.6	70.7
TeG-PS	K400	**94.1**	**71.9**

(a) **Fine-tuning on UCF/HMDB**

method	pre-train data	UCF	HMDB
MemDPC [27]	K400	54.1	30.5
CoCLR [26]	K400	77.8	52.4
CVRL [48]	K400	89.8	58.3
TeG-FG	K400	88.9	60.7
TeG-PS	K400	**91.1**	**64.2**

(b) **Linear evaluation on UCF/HMDB**

method		pre-train data	mAP
Sup. pre-train	R3D-50	K400	19.8
	I3D [10]	ImageNet	22.9
Unsup. pre-train	CVRL [48]	K400	24.1
	VFS [71]	K400	25.9
	TeG-FG	K400	27.7
	TeG-PS	K400	**28.7**

(c) **Fine-tuning on AVA-Kinetics**

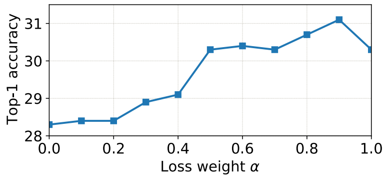
Table 3: **(a, b) Downstream action recognition on UCF101 and HMDB51.** TeG-PS shows competitive performance in fine-tuning and linear evaluation. IN† ImageNet data is used. **(c) Spatiotemporal action localization on AVA-Kinetics.** TeG-PS outperforms its supervised pre-training counterpart by 8.9% mAP using the same R3D-50 backbone, as well as state-of-the-art unsupervised pre-training methods.

AVA-Kinetics [38] provides an important spatiotemporal action localization benchmark for evaluating the learned video features. We use our pre-trained backbone to extract features from the person detections provided by an off-the-shelf detector [75], following the practice in recent work [22, 38]. The results are shown in Tab. 3(c), where TeG-PS achieves 28.7% mAP, outperforming supervised pre-training on Kinetics using the same R3D-50 backbone by a large margin of 8.9% mAP. TeG-PS also shows superior performance when compared with other state-of-the-art unsupervised pre-training methods like CVRL and VFS [71]. TeG-FG is 1.0% mAP lower than TeG-PS. We consider it is reasonable since this task still requires video-level understanding within the proposed regions and learning temporally fine-grained feature across different timestamps inside the video should not be helpful in this case.

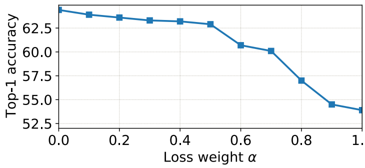
6 Ablation Study

We conduct ablation studies on a few key parameters in our proposed method. We use linear evaluation on VidSitu event classification to justify the performance on temporally fine-grained task and linear evaluation on Kinetics to represent video-level classification task. All experiments are conducted with 200 epochs of pre-training.

Loss weight. Recall that in Equation 3, we propose to use a weight coefficient α to balance the learning of fine-grained and persistent loss. Intuitively, larger α would emphasize more on temporally fine-grained features and suppress the temporal persistency. We ablate the impact of α in Fig. 4. On VidSitu (Fig. 4(a)), we observe that a larger α generally yields better performance as expected except a performance drop when α increases from 0.9 to 1.0. This suggests that completely discarding the temporally persistent learning is not optimal. This is also the reason why we set α as 0.9 instead of 1.0 in TeG-FG. On Kinetics (Fig. 4(b)), we see a consistent drop on the performance as α becomes larger. The reverse trend of performance further enhances our claim that different video tasks require features of different temporal granularities to achieve the best performance. Since we find bringing in temporally fine-grained features is harmful to Kinetics, we focus on VidSitu for the following ablation studies on parameters of temporally fine-grained learning.



(a) **VidSitu event classification**

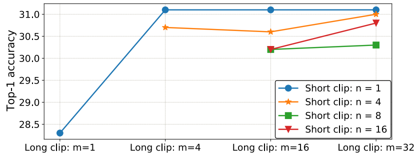


(b) **Kinetics action recognition**

Figure 4: **Ablation on loss weight α.** Performance of features with different granularities specified by α show opposite trends on VidSitu and Kinetics.

clip \ sampling	Random	Contained
Short - Short	27.4	15.2
Long - Short	26.9	**31.1**

(a) Comparison between different sampling strategies. The proposed sampling of a long clip and a containing short clip performs the best.



(b) The choice of n and m in temporal aggregation.

Figure 5: **Sampling strategy and temporal aggregation.** Results are on VidSitu event classification.

Sampling strategy. The proposed sampling strategy requires: 1) two clips to be asymmetric and 2) the short clip being contained within the time duration of the long clip. We ablate on these two design choices in Fig. 5(a). When two clips are both short, random sampling is identical to CVRL [48] and contained sampling losses the diversity in temporal context, thus resulting in poor performance. When two clips are asymmetric, random sampling still does not perform well since the corresponding embeddings between the two clips are inaccurate in the cases that two clips do not have much overlap with each other.

Temporal aggregation. The temporal aggregation parameters m and n determine how dense we want our fine-grained learning to be. We try different combinations of m and n and present their performances in Fig. 5(b). We choose $m = 4, n = 1$ as our default setting due to the simplicity and strong performance.

7 Conclusion

This work studies the impact of temporal granularity in self-supervised video representation learning. We propose a flexible framework named TeG to learn video features of specified temporal granularity and observe that different video tasks require features of different temporal granularities. This insight leads to very competitive results on six video benchmarks. We hope our study can inspire research in video self-supervised learning.

Limitations. From our experiments, we find temporally fine-grained feature performs better on tasks like event classification and boundary detection, while temporally persistent feature shows great advantage on video-level action recognition and spatiotemporal action localization. Manual effort is still needed to find the best recipe for different tasks. Future work could extend TeG to learn a pyramid of representations with coarse to fine temporal granularities from unlabeled videos. The learned representations can therefore be easily transferred to downstream tasks in a more adaptive way.

Acknowledgments. This work was supported in part by the Pioneer Centre for AI, DNRF grant number P1. We would also like to thank the TensorFlow Model Garden team for their infrastructure support and Tsung-Yi Lin for providing valuable feedback.

References

[1] Hassan Akbari, Linagzhe Yuan, Rui Qian, Wei-Hong Chuang, Shih-Fu Chang, Yin Cui, and Boqing Gong. Vatt: Transformers for multimodal self-supervised learning from raw video, audio and text. In *NeurIPS*, 2021.

[2] Jean-Baptiste Alayrac, Adrià Recasens, Rosalia Schneider, Relja Arandjelović, Jason Ramapuram, Jeffrey De Fauw, Lucas Smaira, Sander Dieleman, and Andrew Zisserman. Self-supervised multimodal versatile networks. In *NeurIPS*, 2020.

[3] Humam Alwassel, Dhruv Mahajan, Lorenzo Torresani, Bernard Ghanem, and Du Tran. Self-supervised learning by cross-modal audio-video clustering. In *NeurIPS*, 2020.

[4] Yuki M Asano, Mandela Patrick, Christian Rupprecht, and Andrea Vedaldi. Labelling unlabelled videos from scratch with multi-modal self-supervision. In *NeurIPS*, 2020.

[5] Hamed Azami, Alberto Fernández, and Javier Escudero. Multivariate multiscale dispersion entropy of biomedical times series. *Entropy*, 2019.

[6] Nadine Behrmann, Mohsen Fayyaz, Juergen Gall, and Mehdi Noroozi. Long short view feature decomposition via contrastive video representation learning. In *ICCV*, 2021.

[7] Sagie Benaim, Ariel Ephrat, Oran Lang, Inbar Mosseri, William T Freeman, Michael Rubinstein, Michal Irani, and Tali Dekel. Speednet: Learning the speediness in videos. In *CVPR*, 2020.

[8] Fabian Caba Heilbron, Victor Escorcia, Bernard Ghanem, and Juan Carlos Niebles. Activitynet: A large-scale video benchmark for human activity understanding. In *CVPR*, 2015.

[9] Mathilde Caron, Ishan Misra, Julien Mairal, Priya Goyal, Piotr Bojanowski, and Armand Joulin. Unsupervised learning of visual features by contrasting cluster assignments. In *NeurIPS*, 2020.

[10] Joao Carreira and Andrew Zisserman. Quo vadis, action recognition? a new model and the kinetics dataset. In *CVPR*, 2017.

[11] Brandon Castellano. Video scene cut detection and analysis tool. https://github.com/Breakthrough/PySceneDetect, 2022.

[12] Shixing Chen, Xiaohan Nie, David Fan, Dongqing Zhang, Vimal Bhat, and Raffay Hamid. Shot contrastive self-supervised learning for scene boundary detection. In *CVPR*, 2021.

[13] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. In *ICML*, 2020.

[14] Xinlei Chen, Haoqi Fan, Ross Girshick, and Kaiming He. Improved baselines with momentum contrastive learning. *arXiv preprint arXiv:2003.04297*, 2020.

[15] Madalena Costa, Ary L Goldberger, and C-K Peng. Multiscale entropy analysis of complex physiologic time series. *Physical review letters*, 2002.

[16] Ali Diba, Vivek Sharma, Luc Van Gool, and Rainer Stiefelhagen. Dynamonet: Dynamic action and motion network. In *ICCV*, 2019.

[17] Li Ding and Chenliang Xu. Weakly-supervised action segmentation with iterative soft boundary assignment. In *CVPR*, 2018.

[18] Christoph Feichtenhofer, Haoqi Fan, Bo Xiong, Ross Girshick, and Kaiming He. A large-scale study on unsupervised spatiotemporal representation learning. In *CVPR*, 2021.

[19] Pedro F Felzenszwalb, Ross B Girshick, David McAllester, and Deva Ramanan. Object detection with discriminatively trained part-based models. *TPAMI*, 2009.

[20] Basura Fernando, Hakan Bilen, Efstratios Gavves, and Stephen Gould. Self-supervised video representation learning with odd-one-out networks. In *CVPR*, 2017.

[21] Li Fu, Xiaoxiao Li, Runyu Wang, Zhengchen Zhang, Youzheng Wu, Xiaodong He, and Bowen Zhou. Scala: Supervised contrastive learning for end-to-end automatic speech recognition. *arXiv preprint arXiv:2110.04187*, 2021.

[22] Rohit Girdhar, Joao Carreira, Carl Doersch, and Andrew Zisserman. Video action transformer network. In *CVPR*, 2019.

[23] Ross Girshick, Jeff Donahue, Trevor Darrell, and Jitendra Malik. Rich feature hierarchies for accurate object detection and semantic segmentation. In *CVPR*, 2014.

[24] Jean-Bastien Grill, Florian Strub, Florent Altché, Corentin Tallec, Pierre H Richemond, Elena Buchatskaya, Carl Doersch, Bernardo Avila Pires, Zhaohan Daniel Guo, Mohammad Gheshlaghi Azar, et al. Bootstrap your own latent: A new approach to self-supervised learning. In *NeurIPS*, 2020.

[25] Tengda Han, Weidi Xie, and Andrew Zisserman. Video representation learning by dense predictive coding. In *ICCV Workshops*, 2019.

[26] Tengda Han, Weidi Xie, and Andrew Zisserman. Self-supervised co-training for video representation learning. In *NeurIPS*, 2020.

[27] Tengda Han, Weidi Xie, and Andrew Zisserman. Memory-augmented dense predictive coding for video representation learning. In *ECCV*, 2020.

[28] Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick. Momentum contrast for unsupervised visual representation learning. In *CVPR*, 2020.

[29] Qingqiu Huang, Yu Xiong, Anyi Rao, Jiaze Wang, and Dahua Lin. Movienet: A holistic dataset for movie understanding. In *ECCV*, 2020.

[30] Yu-Gang Jiang, Jingen Liu, Amir R. Zamir, George Toderici, Ivan Laptev, Mubarak Shah, and Rahul Sukthankar. Thumos challenge: Action recognition with a large number of classes. http://crcv.ucf.edu/THUMOS14/, 2014.

[31] Longlong Jing and Yingli Tian. Self-supervised spatiotemporal feature learning by video geometric transformations. *arXiv preprint arXiv:1811.11387*, 2018.

[32] 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.

[33] Dahun Kim, Donghyeon Cho, and In So Kweon. Self-supervised video representation learning with space-time cubic puzzles. In *AAAI*, 2019.

[34] Bruno Korbar, Du Tran, and Lorenzo Torresani. Cooperative learning of audio and video models from self-supervised synchronization. In *NeurIPS*, 2018.

[35] Hildegard Kuehne, Hueihan Jhuang, Estíbaliz Garrote, Tomaso Poggio, and Thomas Serre. Hmdb: a large video database for human motion recognition. In *ICCV*, 2011.

[36] Colin Lea, Austin Reiter, René Vidal, and Gregory D Hager. Segmental spatiotemporal cnns for fine-grained action segmentation. In *ECCV*, 2016.

[37] Hsin-Ying Lee, Jia-Bin Huang, Maneesh Singh, and Ming-Hsuan Yang. Unsupervised representation learning by sorting sequences. In *ICCV*, 2017.

[38] Ang Li, Meghana Thotakuri, David A Ross, João Carreira, Alexander Vostrikov, and Andrew Zisserman. The ava-kinetics localized human actions video dataset. *arXiv preprint arXiv:2005.00214*, 2020.

[39] Rui Li, Yiheng Zhang, Zhaofan Qiu, Ting Yao, Dong Liu, and Tao Mei. Motion-focused contrastive learning of video representations. In *ICCV*, 2021.

[40] Tianwei Lin, Xu Zhao, Haisheng Su, Chongjing Wang, and Ming Yang. Bsn: Boundary sensitive network for temporal action proposal generation. In *ECCV*, 2018.

[41] Tianwei Lin, Xiao Liu, Xin Li, Errui Ding, and Shilei Wen. Bmn: Boundary-matching network for temporal action proposal generation. In *ICCV*, 2019.

[42] Fuchen Long, Ting Yao, Zhaofan Qiu, Xinmei Tian, Jiebo Luo, and Tao Mei. Gaussian temporal awareness networks for action localization. In *CVPR*, 2019.

[43] William Lotter, Gabriel Kreiman, and David Cox. Deep predictive coding networks for video prediction and unsupervised learning. *arXiv preprint arXiv:1605.08104*, 2016.

[44] 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 *CVPR*, 2020.

[45] Alejandro Pardo, Fabian Caba, Juan León Alcázar, Ali K Thabet, and Bernard Ghanem. Learning to cut by watching movies. In *ICCV*, 2021.

[46] Mandela Patrick, Yuki M Asano, Ruth Fong, João F Henriques, Geoffrey Zweig, and Andrea Vedaldi. Multi-modal self-supervision from generalized data transformations. *arXiv preprint arXiv:2003.04298*, 2020.

[47] AJ Piergiovanni, Anelia Angelova, and Michael S. Ryoo. Evolving losses for unsupervised video representation learning. In *CVPR*, 2020.

[48] Rui Qian, Tianjian Meng, Boqing Gong, Ming-Hsuan Yang, Huisheng Wang, Serge Belongie, and Yin Cui. Spatiotemporal contrastive video representation learning. In *CVPR*, 2021.

[49] Adrià Recasens, Pauline Luc, Jean-Baptiste Alayrac, Luyu Wang, Florian Strub, Corentin Tallec, Mateusz Malinowski, Viorica Patraucean, Florent Altché, Michal Valko, et al. Broaden your views for self-supervised video learning. In *ICCV*, 2021.

[50] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. In *NeurIPS*, 2015.

[51] Alexander Richard, Hilde Kuehne, and Juergen Gall. Weakly supervised action learning with rnn based fine-to-coarse modeling. In *CVPR*, 2017.

[52] Arka Sadhu, Tanmay Gupta, Mark Yatskar, Ram Nevatia, and Aniruddha Kembhavi. Visual semantic role labeling for video understanding. In *CVPR*, 2021.

[53] Dian Shao, Yue Zhao, Bo Dai, and Dahua Lin. Finegym: A hierarchical video dataset for fine-grained action understanding. In *CVPR*, 2020.

[54] Dian Shao, Yue Zhao, Bo Dai, and Dahua Lin. Intra-and inter-action understanding via temporal action parsing. In *CVPR*, 2020.

[55] Mike Zheng Shou, Stan Lei, Deepti Ghadiyaram, Weiyao Wang, and Matt Feiszli. Generic event boundary detection: A benchmark for event segmentation. `https://github.com/StanLei52/GEBD/tree/main/eval`, 2021.

[56] Mike Zheng Shou, Stan W Lei, Weiyao Wang, Deepti Ghadiyaram, and Matt Feiszli. Generic event boundary detection: A benchmark for event segmentation. In *ICCV*, 2021.

[57] Zheng Shou, Dongang Wang, and Shih-Fu Chang. Temporal action localization in untrimmed videos via multi-stage cnns. In *CVPR*, 2016.

[58] 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 *CVPR*, 2017.

[59] Panagiotis Sidiropoulos, Vasileios Mezaris, Ioannis Kompatsiaris, Hugo Meinedo, Miguel Bugalho, and Isabel Trancoso. Temporal video segmentation to scenes using high-level audiovisual features. *TCSVT*, 2011.

[60] Ankit Singh, Omprakash Chakraborty, Ashutosh Varshney, Rameswar Panda, Rogerio Feris, Kate Saenko, and Abir Das. Semi-supervised action recognition with temporal contrastive learning. In *CVPR*, 2021.

[61] 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.

[62] Nitish Srivastava, Elman Mansimov, and Ruslan Salakhudinov. Unsupervised learning of video representations using lstms. In *ICML*, 2015.

[63] Chen Sun, Fabien Baradel, Kevin Murphy, and Cordelia Schmid. Learning video representations using contrastive bidirectional transformer. *arXiv preprint arXiv:1906.05743*, 2019.

[64] Chen Sun, Austin Myers, Carl Vondrick, Kevin Murphy, and Cordelia Schmid. Videobert: A joint model for video and language representation learning. In *ICCV*, 2019.

[65] Chen Sun, Arsha Nagrani, Yonglong Tian, and Cordelia Schmid. Composable augmentation encoding for video representation learning. In *ICCV*, 2021.

[66] Jiangliu Wang, Jianbo Jiao, Linchao Bao, Shengfeng He, Yunhui Liu, and Wei Liu. Self-supervised spatio-temporal representation learning for videos by predicting motion and appearance statistics. In *CVPR*, 2019.

[67] Jiangliu Wang, Jianbo Jiao, and Yun-Hui Liu. Self-supervised video representation learning by pace prediction. In *ECCV*, 2020.

[68] Jue Wang, Gedas Bertasius, Du Tran, and Lorenzo Torresani. Long-short temporal contrastive learning of video transformers. *arXiv preprint arXiv:2106.09212*, 2021.

[69] Xiaolong Wang, Ross Girshick, Abhinav Gupta, and Kaiming He. Non-local neural networks. In *CVPR*, 2018.

[70] Dejing Xu, Jun Xiao, Zhou Zhao, Jian Shao, Di Xie, and Yueting Zhuang. Self-supervised spatiotemporal learning via video clip order prediction. In *CVPR*, 2019.

[71] Jiarui Xu and Xiaolong Wang. Rethinking self-supervised correspondence learning: A video frame-level similarity perspective. In *ICCV*, 2021.

[72] Ting Yao, Yiheng Zhang, Zhaofan Qiu, Yingwei Pan, and Tao Mei. Seco: Exploring sequence supervision for unsupervised representation learning. In *AAAI*, 2021.

[73] Hang Zhao, Antonio Torralba, Lorenzo Torresani, and Zhicheng Yan. Hacs: Human action clips and segments dataset for recognition and temporal localization. In *ICCV*, 2019.

[74] Yue Zhao, Yuanjun Xiong, Limin Wang, Zhirong Wu, Xiaoou Tang, and Dahua Lin. Temporal action detection with structured segment networks. In *ICCV*, 2017.

[75] Xingyi Zhou, Dequan Wang, and Philipp Krähenbühl. Objects as points. *arXiv preprint arXiv:1904.07850*, 2019.