TEMPORAL CONTRASTIVE LEARNING WITH CURRICULUM

Shuvendu Roy, Ali Etemad

Queen’s University, Canada
{shuvendu.roy, ali.etemad}@queensu.ca

ABSTRACT

We present ConCur, a contrastive video representation learning method that uses curriculum learning to impose a dynamic sampling strategy in contrastive training. More specifically, ConCur starts the contrastive training with easy positive samples (temporally close and semantically similar clips), and as the training progresses, it increases the temporal span effectively sampling hard positives (temporally away and semantically dissimilar). To learn better context-aware representations, we also propose an auxiliary task of predicting the temporal distance between a positive pair of clips. We conduct extensive experiments on two popular action recognition datasets, UCF101, and HMDB51, on which our proposed method achieves superior performance on video action recognition and video retrieval. Detailed ablation studies show the effectiveness of each of the components of our proposed method.

Index Terms— Action Recognition, Self-supervised Learning, Temporal Representation Learning

1. INTRODUCTION

Self-supervised learning (SSL) has seen tremendous growth in recent years for learning image representations, and has achieved state-of-the-art results in a variety of different downstream applications [1, 2, 3]. Recently a class of self-supervised methods called contrastive learning [1, 4] has shown significant improvement and generalization capabilities across different tasks and domains. The basic idea of contrastive learning is to distinguish between positive and negative samples, where the positives are different views of the same input sample (usually generated by augmentations), while the negatives are derived from different samples.

The success of contrastive learning for video representations has shown a similar trend to that of image representation learning. A number of prior works have directly adopted the popular image-based contrastive methods and applied them to videos with the aid of an additional sampling step for the clips [5, 6]. Unlike image contrastive learning that only applies augmentations to generate two positive samples, videos consist of a temporal dimension from which different sub-clips are sampled and defined as either positive [6] or negatives [7].

The sampling technique of clips and the definition of positive samples is still an open problem for video contrastive learning as different solutions resort to different strategies for this purpose [5, 6]. For example, in [5], sub-clips were sampled from the entire video, and any pair was treated as positive samples. However, it was argued in [6] that frames that are temporally far apart contain totally different contextual information, and thus considering them as positive pairs would be unreasonable. Therefore for deriving the positive pairs, they used a sampling technique that assigned a probability value inversely proportional to the distance between the frames. On the other extreme, distanced clips in the same video were used as negatives in [7]. This issue motivates our paper, where we pose the following questions. (1) How should the positive pairs be defined in contrastive video representation learning? (2) Can positive temporal pairs be selected dynamically (from a temporal perspective) without a fixed definition?

To tackle these problems, we present a method for Contrastive learning using context-aware Curriculum learning (ConCur), a self-supervised approach to learning video representations. Our method samples multiple positive clips from a dynamic temporal span that progressively increases the range, in essence gradually hardening the positive samples. More specifically, at the beginning of the training, ConCur samples positive clips that are temporally overlapped, thus containing similar semantic contexts. As the training progresses, we increase the temporal span from which positive samples are randomly sampled, effectively increasing the probability of sampling positives that are temporally far apart and semantically dissimilar. We adopt a modified Multi-instance MoCo (MI-MoCo) [4] as the contrastive loss for our proposed method. To learn better temporal representations, we also propose a Context Similarity loss that facilitates the
learning of context-aware video representations by predicting the temporal distance between two positive clips. We rigorously evaluate ConCur on widely used public datasets for two downstream tasks, namely activity recognition and video retrieval, and show that our method achieves superior results on multiple video encoders (e.g., R(2+1)D, C3D). An overview of ConCur is shown in Fig. 1.

We make the following contributions in this paper. (1) We introduce a temporal sampling strategy for sampling positive clips with a dynamic temporal span. We show that randomly sampling positives over the entire duration of a video hurt the representation learnt by contrastive loss. (2) We propose a curriculum learning strategy for contrastive video learning that increases the temporal span from which positive clips are randomly sampled, effectively increasing the hardness for the contrastive loss over a given period. The proposed curriculum learning module increases the accuracy of the downstream application with no increase in model size or FLOPs. (3) We also propose a Context Similarity loss term to learn better temporal representation by predicting the temporal distance between the two positive clips.

2. PROPOSED METHOD

Preliminaries. The fundamental concept of our proposed method is based on contrastive learning [1], which aims to learn from positive (usually generated from augmentations) and negative samples (all other samples). More specifically, it learns by bringing the embedding of positive samples closer and pushing the negatives apart from the positives. For an input video \(v\), a 3D CNN encoder with a projection head (linear layers with non-linearity) is used to extract a query embedding, \(q = f_0(v)\). From an augmentation \(v' = aug(v)\), a momentum encoder (moving average of encoder) generates a positive key embedding, \(k^+ = f_{\alpha_m}(v')\). A dictionary stores the keys from the previous iterations of training, which are used as negatives \(k^-\). Given an encoder query \(q\), positive key \(k^+\), and negative keys \(k^- = \{k_0, k_1, \ldots\}\), the contrastive loss function can be written as:

\[
L_q = -\log \frac{\exp(sim(q, k^+)/\alpha)}{\sum_{k \in (k^-)} \exp(sim(q, k)/\alpha)},
\]

where \(\alpha\) is a temperature parameter and \(sim(q, k)\) is the cosine similarity represented as \(sim(q, k) = (q^T k) / (||q|| \cdot ||k||)\).

Multi-instance Sampling with a Temporal Span. For a reasonably long video, the semantic context at the start of a video can be dissimilar to the end of that video. This is why sampling positive clips over the entire duration of a video and encouraging their embeddings to be similar is not reasonable. To tackle this problem, we propose a sampling method that imposes a constraint on the temporal span from which positive clips are sampled. The proposed sampling technique only picks positive samples from the defined temporal span, and any clip sampled from a different video is considered negative. To further improve representation learning, we adopt multi-instance contrastive learning, which uses multiple positive keys instead of one. Given a video clip with a temporal length (number of frames) \(T\), and a temporal span \((TS)\), the method first picks a random starting frame \(s \in [1, T - TS]\). Accordingly, the temporal window for positives is defined as \(w \in [s, s + TS]\). To facilitate multi-instance contrastive learning, we then sample \(\rho\) clips of \(T\) consecutive frames from the sampling window \(w\) to get \(\rho\) positive samples \(x^+ = \{x^+_1, x^+_2, \ldots, x^+_\rho\}\). Following [4], we adopt a symmetric multi-instance loss, where embedding of each clip in \(x^+\) is considered as query and \(\rho - 1\) other clips are considered as positive keys \(\{k^+\}\). The modified loss for Multi-instance Contrastive loss is represented as follows (the total loss is averaged over \(\rho\)):

\[
L_{MI} = -\log \frac{\sum_{k \in (k^+)} \exp(sim(q, k^+)/\alpha)}{\sum_{k \in (k^+, k^-)} \exp(sim(q, k)/\alpha)}.
\]

In the following sub-section, we describe how we use curriculum learning to progressively update \(TS\).

Curriculum Learning for Temporal Span Update. Curriculum learning [8] is a paradigm that is inspired by the human learning behaviour of staring with ‘easy’ concepts and slowly learning more complex topics. In video contrastive settings, if a pair of sampled clips \((x^i_1, x^i_2)\) are very close or overlap with one another, they are more likely to contain semantically similar content. We define pairs that are temporally close as ‘easy’ positive pairs. On the other hand, pairs \((x^a_1, x^a_2)\) that are temporally far apart are considered ‘hard’ positives. Here, we propose a curriculum approach that gradually increases the temporal span, \(TS\), of positive samples over the training epochs, effectively hardening the positive samples for our contrastive loss. The proposed curriculum learning component is illustrated in Fig. 2 (left), where the training starts with a short temporal span and is then increased over the training iterations. We investigate the proposed component in two settings. In the first setting, we increase the hardness over the entire training phase. In the second setting, we limit the number of epochs over which hardening occurs \((E_{CL})\), and a constant hardness is used beyond that threshold (see Fig. 2 (right)). The temporal span at a given epoch \(e\) is formulated as:

\[
TS_e = \min \left( TS_m, TS_i + \left( \frac{TS_m - TS_i}{E_{CL}} \right) \times e \right),
\]

where \(TS_m\) is the maximum temporal span, \(TS_i\) is the initial temporal span, and \(E_{CL}\) is the total number of epochs over which hardening is performed.

Context Similarity. To learn context-aware representation, we also propose an auxiliary task of predicting the temporal distance (number of frames) between any two clips. This helps the model learn better contextual information regarding the location of the input samples. To predict the distance between a query embedding \(q\) and a key embedding \(k\), we add a single linear layer that takes the concatenation of \(q\) and \(k\) as
work and trained with 8 Nvidia V100 GPUs.

**Sensitivity Study.** First, we conduct a sensitivity study for different temporal spans (TS) for sampling the positives presented in Fig. 4 (left). These studies were done on the K200 dataset and trained for 100 epochs. As we observe, the performance of the model increases when we increase the temporal span from 32 to 100. However, the accuracy drops when we use larger values. For example, a temporal span of 125 performs worse than 32. The accuracy is the lowest when the length of the full video is used as the temporal span. The reason behind this is likely the significant semantic contextual difference between the first and last parts of a long video. This verifies the fact that sampling from the full video is not optimal for contrastive learning. For the rest of the experiments, we set TS = 100.

Next, we experiment with different ECL values to identify its impact. We present the results in Fig. 4 (right) where we observe that ECL = 50 gives the best results, followed closely by 25 and 75. However, curriculum learning over a very short period (e.g., ECL = 10) do not result in any notable improvements compared to not using curriculum learning. Moreover, large ECL values also do not help the model. This is in line with our expectation of identifying an optimum level of hardness, where harder or simpler self-supervisory signals hurt the performance [27].

**Ablation Experiments.** To investigate the impact of each of the proposed components in this method, in Table 1 we present a detailed ablation study. As we observe from the table, removing the CS loss from ConCur reduces the accuracy by around 0.6%, which shows the importance of the context similarity loss in the proposed method. It should be noted that in recent years, improvements made by the state-of-the-art are in a similar range, often improving prior works by approximately 0.5% to 2.0%. Next, we remove the curriculum learning component, effectively allowing the positives to be sampled from the entire duration of the video, which shows a drop of 0.9% accuracy on the downstream task. Finally, we train the model by removing both CL loss and curriculum learning and observe a drop of 1.6% accuracy. This shows the effec-

---

**Table 1:** Ablation results on UCF101, pre-trained on K200.

| Curriculum | CS loss | Accuracy (%) |
|------------|---------|--------------|
| ✓          | ✓       | 81.08        |
| ✓          | ✗       | 80.64        |
| ✗          | ✓       | 80.17        |
| ✗          | ✗       | 79.46        |

---

**Fig. 3:** Visual illustration of the proposed model.

Let $\phi(x)$ be a function that returns the frame number at which a positive clip $x$ starts within the original video. We define the distance between the query clip $x_i^t$ and any positive key clip $x_{k}^t$ as $d_k = abs(\phi(x_i^t) - \phi(x_{k}^t))$, and the predicted distance by the model as $d'_k$. For $k$ positive keys in the proposed setting, we define the context similarity loss as:

$$L_{CS} = \frac{1}{k+1} \sum_{k \in \{k+1\}} (d_k - d'_k)^2 / \rho. \quad (4)$$

Accordingly, the overall loss for ConCur is defined as:

$$L_{ConCur} = L_{MI} + L_{CS}. \quad (5)$$

The overall diagram is illustrated in Fig. 3.

---

**3. EXPERIMENTS AND RESULTS**

**Implementation Details.** Following the convention of existing literature [5, 9], we pre-train our model with Kinetics400 [10] and Kinetics200 [11], and fine-tune on UCF101 [12] and HMDB51 [13] datasets. The pre-training is done with a batch-size of 16 clips per GPU for 400 epochs. Momentum SGD is used for the training with a momentum value of 0.9. Following [1], a temperature value of 0.0001. We use $t = 16$ as the number of frames with a spatial resolution of $s = 112$ for the input to the model. The clips are selected after the original videos are re-sampled to 15 fps. For the momentum encoder, we use the key dictionary of size 65536, and a momentum value of 0.999. Following the pre-training step, we fine-tune the model with the pre-trained encoder on UCF101 and HMDB51 datasets. In this stage, we remove the projection head and add a randomly initialized linear classification layer. The full model is then fine-tuned for 100 epochs. The model is implemented with PyTorch frame-
Table 2: Accuracy with different self-supervised methods.

| Method      | Backbone | w/o ConCur | w/ ConCur |
|-------------|----------|------------|-----------|
| BYVE [14]   | C3D      | 79.02      | 80.55     |
| SimSiam [15]| R(2+1)D  | 78.63      | 80.13     |
| Barlow Twins [16] | R(2+1)D | 78.48      | 79.50     |
| VICReg [17] | R(2+1)D  | 78.80      | 80.04     |

Table 3: Comparison with prior works.

| Method          | Backbone | Pre-train | Fine-tune | Res. | Frames | UCF-101 (%) | HMDB-51 (%) |
|-----------------|----------|-----------|-----------|------|--------|------------|------------|
| RTT [18]        | C3D      | ✓         | ✓         | 112  | 16     | 68.5       | 38.4       |
| RTT [18]        | R(2+1)D  | ✓         | ✓         | 112  | 16     | 69.9       | 39.6       |
| PRP [20]        | C3D      | ✓         | ✓         | 112  | 16     | 69.1       | 34.5       |
| VCP [21]        | C3D      | ✓         | ✓         | 112  | 16     | 68.5       | 32.5       |
| VCP [21]        | R(2+1)D  | ✓         | ✓         | 112  | 16     | 65.6       | 28.4       |
| Var. SP [23]    | C3D      | ✓         | ✓         | 112  | 16     | 70.4       | 34.3       |
| MoCo v2 [24]    | R(2+1)D  | ✓         | ✓         | 112  | 16     | 67.5       | 14.6       |
| Ours            | R(2+1)D  | ✓         | ✓         | 112  | 16     | 67.9       | 43.0       |
| Ours            | C3D      | ✓         | ✓         | 112  | 16     | 72.9       | 48.2       |
| Ours            | R(2+1)D  | ✓         | ✓         | 112  | 16     | 67.7       | 32.2       |

Table 4: Recall at top-K. Comparison with prior work for video retrieval task on UCF-101 and HMDB-51.

| Method          | Backbone | Pre-train | Fine-tune | Res. | Frames | UCF-101 (%) | HMDB-51 (%) |
|-----------------|----------|-----------|-----------|------|--------|------------|------------|
| VCP [22]        | C3D      | ✓         | ✓         | 112  | 16     | 66.6       | 32.2       |
| VCP [22]        | R(2+1)D  | ✓         | ✓         | 112  | 16     | 71.2       | 35.0       |
| VCP [22]        | C3D      | ✓         | ✓         | 112  | 16     | 72.4       | 30.9       |
| PRP [20]        | R(2+1)D  | ✓         | ✓         | 112  | 16     | 74.8       | 39.6       |
| PacePred [26]   | C3D      | ✓         | ✓         | 112  | 16     | 75.0       | 35.9       |
| PacePred [26]   | R(2+1)D  | ✓         | ✓         | 112  | 16     | 75.9       | 34.4       |
| Ours            | R(2+1)D  | ✓         | ✓         | 112  | 16     | 72.1       | 41.6       |
| Ours            | C3D      | ✓         | ✓         | 112  | 16     | 77.9       | 48.2       |
| Ours            | R(2+1)D  | ✓         | ✓         | 112  | 16     | 66.6       | 12.2       |

4. CONCLUSIONS

We present a contrastive video representation learning method that uses curriculum learning for selecting the positive samples used by the contrastive loss. ConCur starts the contrastive training with easy positive samples which are temporally close and semantically similar, and progressively samples harder positives that are temporally away and semantically dissimilar. The experiments conducted in this paper show that our method improves performance versus blind sampling of positives from the entire input video. We also show that there is an optimal level of difficulty where the performance is maximum. To learn better context-aware representations, we also propose the auxiliary task of predicting the temporal distance between a positive pair of clips. Ablation studies show the effectiveness of each of the components of our method. We achieve superior performance with two benchmark datasets on video action recognition and video retrieval using two different encoders.

Acknowledgements

We would like to thank BMO Bank of Montreal and Mitacs for funding this research. We are also thankful to SciNet HPC Consortium for helping with the computation resources.
5. REFERENCES

[1] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton, “A simple framework for contrastive learning of visual representations,” in Proc. ICML, 2020, pp. 1597–1607.

[2] Shuvendu Roy and Ali Etemad, “Self-supervised contrastive learning of multi-view facial expressions,” in Proc. ICMI, 2021, pp. 253–257.

[3] Shuvendu Roy and Ali Etemad, “Spatiotemporal contrastive learning of facial expressions in videos,” in Proc. ACII, 2021, pp. 1–8.

[4] Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick, “Momentum contrast for unsupervised visual representation learning,” in Proc. CVPR, 2020, pp. 9729–9738.

[5] Christoph Feichtenhofer, Haoqi Fan, Bo Xiong, Ross Girshick, and Kaiming He, “A large-scale study on unsupervised spatiotemporal representation learning,” in Proc. CVPR, 2021, pp. 3299–3309.

[6] Rui Qian, Tianjian Meng, Boqing Gong, Ming-Hsuan Yang, Huisheng Wang, Serge Belongie, and Yin Cui, “Spatiotemporal video representation learning,” in Proc. CVPR, 2021, pp. 6964–6974.

[7] Pierre Sermanet, Corey Lynch, Yevgen Chebotar, Jasmine Hsu, Eric Jang, Stefan Schaal, Sergey Levine, and Google Brain, “Time-contrastive networks: Self-supervised learning from video,” in Proc. ICRA, 2018, pp. 1134–1141.

[8] Joshua Bengio, Jérôme Louradour, Ronan Collobert, and Jason Weston, “Curriculum learning,” in Proc. ICML, 2009, pp. 41–48.

[9] Tian Pan, Yibing Song, Tianyu Yang, Wenhai Jiang, and Wei Liu, “Videomoco: Contrastive video representation learning with temporally adversarial examples,” in Proc. CVPR, 2021, pp. 11205–11214.

[10] 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:1705.06950, 2017.

[11] Saining Xie, Chen Sun, Jonathan Huang, Zhuowen Tu, and Kevin Murphy, “Rethinking spatiotemporal feature learning for video understanding,” arXiv:1712.04851, 2017.

[12] 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.

[13] Hildegard Kuehne, Huihuan Jhuang, Estibaliz Garrote, Tomaso Poggio, and Thomas Serre, “Hmdb: a large video database for human motion recognition,” in Proc. ICCV, 2011, pp. 2556–2563.

[14] Jean-Bastien Grill, Florian Strub, Florent Altché, Corentin Tallec, Pierre Richemond, Elena Buchatskaya, Carl Doersch, Bernardo Avila Pires, Zhaohan Guo, Mohammad Gheshlaghi Azar, et al., “Bootstrap your own latent-a new approach to self-supervised learning.” in Proc. NeurIPS, 2020, pp. 21271–21284.

[15] Xinlei Chen and Kaiming He, “Exploring simple siamese representation learning,” in Proc. CVPR, 2021, pp. 15750–15758.

[16] Jure Zbontar, Li Jing, Ishan Misra, Yann LeCun, and Stéphane Deny, “Barlow twins: Self-supervised learning via redundancy reduction,” in Proc. ICML, 2021, pp. 12310–12320.

[17] Adrien Bardes, Jean Ponce, and Yann LeCun, “Vi-creg: Variance-invariance-covariance regularization for self-supervised learning,” arXiv preprint arXiv:2105.04906, 2021.

[18] Simon Jenni, Givi Meishvili, and Paolo Favaro, “Video representation learning by recognizing temporal transformations,” in Proc. ECCV, 2020, pp. 425–442.

[19] Rui Qian, Yuxi Li, Huabin Liu, John See, Shuangrui Ding, Xian Liu, Dian Li, and Weiyao Lin, “Enhancing self-supervised video representation learning via multi-level feature optimization,” arXiv preprint arXiv:2108.02183, 2021.

[20] Yuan Yao, Chang Liu, Dezhuo Luo, Yu Zhou, and Qixiang Ye, “Video playback rate perception for self-supervised spatio-temporal representation learning,” in Proc. CVPR, 2020, pp. 6548–6557.

[21] Dezhuo Luo, Chang Liu, Yu Zhou, Dongbao Yang, Can Ma, Qixiang Ye, and Weiping Wang, “Video cloze procedure for self-supervised spatio-temporal learning,” in Proc. AAAI, 2020, pp. 11701–11708.

[22] Dejing Xu, Jun Xiao, Zhou Zhao, Jian Shao, Di Xie, and Yueting Zhuang, “Self-supervised spatiotemporal learning via video clip order prediction,” in Proc. CVPR, 2019, pp. 10334–10343.

[23] Hyeon Cho, Taehoon Kim, Hyung Jin Chang, and Wonjun Hwang, “Self-supervised spatio-temporal representation learning using variable playback speed prediction,” arXiv:2003.02692, 2020.

[24] Jinpeng Wang, Yuting Gao, Ke Li, Yiqi Lin, Andy J Ma, Hao Cheng, Pai Peng, Feiyue Huang, Rongrong Ji, and Xing Sun, “Removing the background by adding the background: Towards background robust self-supervised video representation learning,” in Proc. CVPR, 2021, pp. 11804–11813.

[25] Peihao Chen, Deng Huang, Dongliang He, Xiang Long, Runhaoh Zeng, Shilei Wen, Mingkui Tan, and Chuang Gan, “Rspnet: Relative speed perception for unsupervised video representation learning,” in Proc. AAAI, 2021, pp. 1045–1053.

[26] Jiangluo Wang, Jianbo Jiao, and Yun-Hui Liu, “Self-supervised video representation learning by pace prediction,” in Proc. ECCV, 2020, pp. 504–521.

[27] Srikar Appalaraju, Yi Zhu, Yusheng Xie, and István Fehérvári, “Towards good practices in self-supervised representation learning,” arXiv:2012.00868, 2020.