Speed Co-Augmentation for Unsupervised Audio-Visual Pre-training

Jiangliu Wang
The Chinese University of Hong Kong

Jianbo Jiao
University of Birmingham

Yibing Song
Fudan University

Stephen James
Dyson Robot Learning Lab

Zhan Tong
Tencent AI Lab

Chongjian Ge
The University of Hong Kong

Pieter Abbeel
UC Berkeley

Yun-Hui Liu
The Chinese University of Hong Kong

Abstract

This work aims to improve unsupervised audio-visual pre-training. Inspired by the efficacy of data augmentation in visual contrastive learning, we propose a novel speed co-augmentation method that randomly changes the playback speeds of both audio and video data. Despite its simplicity, the speed co-augmentation method possesses two compelling attributes: (1) it increases the diversity of audio-visual pairs and doubles the size of negative pairs, resulting in a significant enhancement in the learned representations, and (2) it changes the strict correlation between audio-visual pairs but introduces a partial relationship between the augmented pairs, which is modeled by our proposed SoftInfoNCE loss[1] to further boost the performance. Experimental results show that the proposed method significantly improves the learned representations when compared to vanilla audio-visual contrastive learning.

1. Introduction

Audio-visual contrastive learning [14, 17] for unsupervised pre-training has received growing attention due to the observation that video content is usually accompanied by audio signals. The alignment of signals across audio and video forms a natural correspondence to benefit contrastive learning. Under this framework, quite a few existing approaches [13, 17] focus on achieving better discrimination of positive and negative pairs to improve audio-visual representation learning. While promising results have been achieved, most works [12, 13] apply data augmentations to
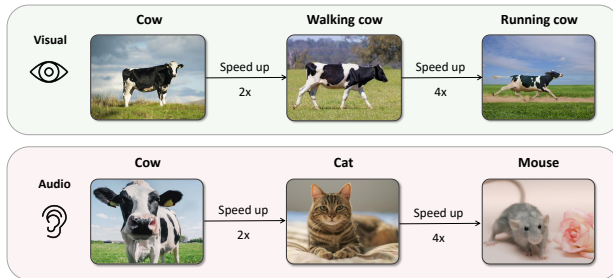

Figure 1. An intuitive example of *semantic shift* after applying speed co-augmentation on audio-visual pairs. Top: when speeding up video data, the semantic meaning of the content doesn't change drastically. Bottom: when speeding up audio data, the semantic meaning of the content changed drastically.

each modality individually, which may potentially limit the diversity of the generated data views and restrict the potential of augmentation for contrastive learning.

In this work, we propose a novel technique termed "speed co-augmentation" for unsupervised audio-visual pre-training, which involves modifying the playback speeds of both audio and visual data simultaneously. The speed co-augmentation method enhances the diversity of audio-visual pairs and doubles the number of negative pairs during training, which has been shown to be a crucial aspect of contrastive learning [5]. Our experimental results demonstrate that this simple co-augmentation method yields a significant performance improvement of 10.0% over the baseline audio-visual contrastive learning approach on the HMDB51 [10] dataset.

Meanwhile, it was observed that after the speed co-augmentation, the audio and video pairs derived from the same clip are no longer strictly positively related, as is com-

---

[1] Our study [6] also validates the effectiveness of the proposed "SoftInfoNCE loss" in single-modality contrastive learning.

monly assumed. As an intuitive special example (Fig. 1), a sped-up *cow* still visually looks like a *cow*, but a sped-up *cow* may auditorily sound like a *cat*. To generalize, we posit that there exists a partial relationship between the augmented audio-visual pairs, which is influenced by the degree of speed augmentation applied. To capture this relationship, we introduce a cross-affinity module that automatically learns the audio-visual correlations across different views. The resulting learned correlations quantitatively measure the audio-visual consistency and are employed for computations of SoftInfoNCE loss, leading to a further performance boost. Combining the proposed speed augmentation and the cross-affinity module, we present a Speed-augmented visual-audio Contrastive Learning framework, which we call *SvaCLR*.

2. Method

Our target is to train video and audio encoders via unsupervised contrastive learning. Given an aligned pair (v, a), we apply speed-up augmentations on both audio a and video v data to synthesize two additional views (*i.e.*, $\widetilde{v}$ and $\widetilde{a}$). These audio and video samples are then fed into the audio and video encoders $f(\cdot)$ and $g(\cdot)$ to extract representations y. We then project the video and audio representations separately via projectors $h_v(\cdot)$ and $h_a(\cdot)$. The projected embeddings z are then utilized to compute the contrastive InfoNCE loss [15]. In parallel, we introduce a cross-affinity module to model the audio-visual embedding correlations. The modeled correlations are used to reweigh the InfoNCE loss, *i.e.*, the proposed SoftInfoNCE loss, when learning audio-visual representations. In the following, we first introduce the speed-up augmentation with the vanilla InfoNCE loss and then introduce the cross-affinity module to re-weigh the InfoNCE loss (*i.e.*, SoftInfoNCE loss).

2.1. Speed co-augmentation

For speed co-augmentation, we use a speed library to diversify training data pairs. We use $\mathcal{T}$ to represent the speed co-augmentation set in which the maximum speed is denoted by S. Each time, two speed augmentation factors for the audio and video data are selected randomly from $\mathcal{T}$ and are applied to each data, respectively. In practice, the proposed speed co-augmentation is implemented by applying different sampling rates of the audio and video samples.

Before computing the contrastive InfoNCE loss [15], we project the video and audio representations separately via projectors. We use one video projector $h_v(\cdot)$ to connect the video encoder and use one audio projector $h_a(\cdot)$ to connect the audio encoder. The projected representations are then utilized to compute the contrastive InfoNCE loss as follows:
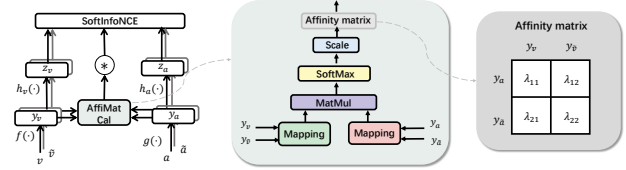

Figure 2. The proposed cross-affinity module for SoftInfoNCE loss computation. The cross-modality attention module takes video and audio representations as input where there are co-augmented audio-visual data. The output is a cross-modality affinity matrix shown on the right. Each element in this matrix represents the correlations between audio and video for each input signal.

$$L(i, j) = \frac{\exp(z_i \cdot z_j / \eta)}{\exp(z_i \cdot z_j / \eta) + \sum_{\substack{j=1 \\ j \neq i}}^{N} \exp(z_i \cdot z_j / \eta)} \quad (1)$$

where $z_i = h_a(y_i)$ is the audio projection, $z_j = h_v(y_j)$ is the video projection, and η is a constant temperature value. The dot product measures the similarity between the projected audio and video representations. For the input audio a_i, the summation term is computed by utilizing all the video clips v_j, as long as a_i and v_j are from different samples (*i.e.*, unpaired).

2.2. Cross-affinity module

We propose a cross-affinity module to measure the correlations between the augmented video and audio representations. Fig. 2 illustrates the proposed module. Given the audio embedding y_i and the video embedding y_j, the cross-modality attention $\lambda(a_i^{\tau_1}, v_j^{\tau_2})$ can be computed as follows:

$$\lambda(a_i^{\tau_1}, v_j^{\tau_2}) = \text{softmax}\left[l(y_i) \times l(y_j^{\mathsf{T}}) \right], \quad (2)$$

where $l(\cdot)$ is a mapping with learnable parameters. The projected video and audio embeddings are correlated via the matrix multiplication operation. In practice, three different mapping function are examined, including identity mapping, linear mapping and nonlinear mapping, from those we find the identity mapping achieves the best results. We speculate this is because heavier mapping could deteriorate the ability of encoders to learn general representations.

We compute the cross-modality attention in Eq. 2 for one co-augmented audio-visual view. The cross-modality affinity can be formulated as a two-by-two matrix (as shown in Fig. 2 right). Each element in this matrix represents the correlation between the speed-augmented audio and video views. By using these elements, we reweigh the contributions of each co-augmented audio-visual view when computing the contrastive loss.

Table 1. Effectiveness of the proposed speed co-augmentation and cross-affinity module. Speed $s = [a, b]$ represents that the lower bound speed is a while the upper bound is b.

Pre-training Experimental Setup				Downstream Acc.	
speed	Re-weight	Speed	Loss	HMDB51	K-sounds@1
✗	✗	-	InfoNCE	54.2	2.6
✓	✗	$s = [1, 2]$	InfoNCE	63.8 (+9.6)	3.1 (+0.5)
✓	✗	$s = [1, 4]$	InfoNCE	<u>65.1</u> (+10.9)	<u>3.5</u> (+0.9)
✓	✗	$s = [1, 6]$	InfoNCE	64.2 (+10.0)	3.2 (+0.6)
✓	✓	$s = [1, 4]$	**SoftInfoNCE**	**66.1 (+11.9)**	**4.6 (+2.0)**

2.3. Training with SoftInfoNCE

Following [12, 17], we use a 9-layered 2D ResNet [7] as the audio encoder and R(2+1)D-18 [19] as the video encoder. The projector is a two-layered multilayer perceptron (MLP). The training process is end-to-end, without using a two-stage setting as in previous works [13, 14].

Given a batch of audio-visual pairs $\mathcal{A}$ and $\mathcal{V}$, where both $\mathcal{A}$ and $\mathcal{V}$ contain N samples, we denote the speed-up augmentation set as $\mathcal{T}$, from which we can sample augmentations $\tau \sim p(\mathcal{T})$. The encoders and projectors are trained with the following SoftInfoNCE loss:

$$\mathcal{L}(f, g, \mathcal{A}, \mathcal{V}) = \mathbb{E}_{(\tau_1, \tau_2) \sim p(\mathcal{T})} \left[\frac{1}{N^2} \sum_{i=1}^{N} \sum_{j=1}^{N} \lambda(a_i^{\tau_1}, v_j^{\tau_2}) \cdot L(g(a_i^{\tau_1}), f(v_j^{\tau_2})) \right] \quad (3)$$

where $L(\cdot, \cdot)$ is the contrastive InfoNCE loss function as illustrated in Eq. 1, $a_i \in \mathcal{A}$, and $v_j \in \mathcal{V}$. The cross-modality attention $\lambda(\cdot, \cdot)$ takes the audio and video signal as input and measures their correlations. The output correlation value further reweighs the contrastive loss during the training process consequently.

3. Experiments

3.1. Implementation details

We use Kinetics-Sounds (K-Sounds) [2], Kinetics-400 (K400) [8], and VGGSound [4] as the pre-training datasets to evaluate the effectiveness of our proposed approach.

For pre-training, we use Stochastic Gradient Descent (SGD) as the optimizer. We use 10 epochs to warm up the learning rate from 64×10^{-3} to 64×10^{-2} and then using a cosine learning rate decay to 10×10^{-2} in the remaining 90 epochs. Training is done on 64 V100 GPUs with a mini-bath size of 8 on each, resulting in total batch size of 512. The total training time is around 30 hours for 100 epochs.

For fine-tuning, we follow the fine-tuning setting of GDT [17]. SGD is used as the optimizer and the initial learning rate is set to 2.5×10^{-3}, where it is warmed up to 2×10^{-2} in the first two epochs, and decreased by 5×10^{-2} at 6 and 10 epochs. Training is stopped at 12 epochs.
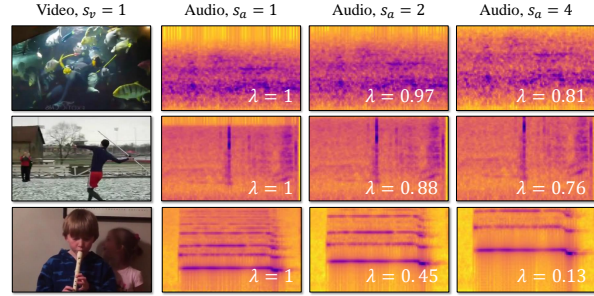

Figure 3. Examples of the visual-audio pairs with speed co-augmentation and their partial relationship weights. For each sample from left to right: one frame from the video clip, spectrogram of the corresponding audio, spectrogram of the augmented audio with speed 2, and spectrogram of the augmented audio with speed 4. λ reflects the degree of correlations learned by our proposed cross-affinity module.

3.2. Ablation studies

Speed co-augmentation. We investigate different augmentations for audio and video data from the same speed distribution in Table 1. It can be seen that: (1) compared to the no speed-up augmentation, using only two speed candidates can already significantly improve the action recognition performance from 54.2% to 63.8%. This validates the effectiveness of the proposed speed-up augmentation. (2) When audio-video pairs are sped up from the same distribution, best action recognition accuracy is achieved when $s = [1, 4]$. Both $s = [1, 2]$ and $s = [1, 6]$ settings perform inferior than the $s = [1, 4]$ setting. We suspect that this is because compared to $s = [1, 2]$, using $s = [1, 4]$ provides more view synthesis while $s = [1, 6]$ could be a bit difficult for the network to learn useful semantic representations.

Cross-affinity module. Based on the best speed augmentation setting $s_a = s_v = [1, 4]$, we investigate the effectiveness of the proposed cross-affinity module in Table 1. It can be seen that with the proposed cross-affinity module and the SoftInfoNCE loss design, the performances can be further improved significantly (*e.g.*, +11.9% for HMDB51). To better understand what the cross-affinity module learns, we visualize the visual-audio pairs with the speed augmentation and their corresponding relationship weights in Fig. 3. It can be seen that videos with manifested audio signals are more sensitive to the proposed speed augmentation, where the weight between the video and augmented audio is relatively small after the speed transformation.

3.3. Comparison with previous works

Action recognition. We compare our proposed SvaCLR with other approaches in Table 2. From the results we can see that: (1) When pre-trained on a medium-scale dataset, K-Sounds, our approach significantly outperforms previous

Table 2. Comparison with previous approaches on UCF101 [18] and HMDB51 [10] datasets.

Method	Pre-training Experimental Setup			Downstream Acc.	
	Architecture	Dataset (duration)	Resolution	HMDB51	UCF101
ACC [12]	R3D-18	K-Sounds (2d)	224×224	40.6	77.2
SvaCLR (Ours)	R3D-18	K-Sounds (2d)	128×128	**54.9**	**86.4**
L^3-Net [2]	VGG-16	K400 (28d)	-	40.2	72.3
PEMT [11]	SlowFast	K400 (28d)	128×128	-	85.2
GDT [17]	R(2+1)D-18	K400 (28d)	128×128	62.3	90.9
SvaCLR (Ours)	R(2+1)D-18	K400 (28d)	128×128	66.1	91.5
SvaCLR (Ours)	R(2+1)D-18	VGGSound (23d)	128×128	**67.2**	**92.0**
Multisensory [16]	R3D-18	K400 (28d)	224×224	-	82.1
SeLaVi [3]	R(2+1)D-18	K400 (28d)	224×224	47.1	84.2
XDC [1]	R(2+1)D-18	K400 (28d)	224×224	52.6	86.2
AVTS [9]	MC3-18	K400 (28d)	224×224	56.9	85.8
STiCA [17]	R(2+1)D-18	K400 (28d)	224×224	60.5	-
AVID [14]	R(2+1)D-18	K400 (28d)	224×224	60.8	87.5
ACC [12]	R3D-18	K400 (28d)	224×224	61.8	90.2
SvaCLR (Ours)	R(2+1)D-18	K400 (28d)	224×224	66.8	92.2
SvaCLR (Ours)	R(2+1)D-18	VGGSound (23d)	224×224	**67.8**	**92.6**

works. We improve performances on UCF101 and HMDB51 datasets by large margins, 9.2% and 14.3%. This demonstrates that our proposed co-augmentation method enlarges the diversity of the training views and benefits contrastive learning a lot. **(2)** Our approach demonstrates great scalability in terms of dataset size. When pre-trained on a large dataset K400, our approach exceeds the state-of-the-art audio-visual representation learning approach GDT [17] by a large margin, especially on the HMDB51 dataset, where we outperform GDT by 3.8%. Note that GDT applies hierarchical data augmentations while we only use one-speed augmentation. **(3)** Our approach also demonstrates scalability in terms of resolution. We can further improve the performances by using a large input size.

Audio-Video Retrieval. To further evaluate the cross-modality ability of the proposed approach, we propose to use an audio-video retrieval task on K-Sounds [2]. We compare to audio-visual contrastive learning with vanilla InfoNCE loss and current state-of-the-art GDT [17] in Table 3. We show that our approach achieves the best performances on both audio-to-video retrieval task and video-to-audio retrieval task. It is interesting to note that pre-trained on K-sounds can achieve better performance to retrieve top-1 nearest neighbor. But its ability to generalize to more visual-audio pairs is restricted that it performs worse than pre-training on a larger dataset K400 to retrieve top-5, top-10, and top-20 nearest neighbors.

4. Conclusions

We proposed a speed co-augmentation method for unsupervised audio-visual pre-training. We observed that speed co-augmentation leads to a partial relationship between audio-visual pairs. To combat this, we propose a

Table 3. Comparison with other approaches on video-audio retrieval task. "Baseline" represents the vanilla audio-video contrastive learning with InfoNCE loss.

Method	Pre-train Dataset	Video $\rightarrow$ Audio				Audio $\rightarrow$ Video			
		R@1	R@5	R@10	R@20	R@1	R@5	R@10	R@20
Baseline	K400	2.6	13.8	26.1	44.3	3.6	13.7	25.3	41.0
GDT [17]	K400	3.1	14.3	24.6	40.8	3.6	16.1	26.4	43.9
SvaCLR (Ours)	K-Sounds	**5.1**	14.1	25.4	42.3	**4.7**	16.0	27.2	43.6
SvaCLR (Ours)	K400	4.6	**17.2**	**28.3**	**44.6**	4.2	**16.4**	**27.6**	**44.3**

cross-affinity module, which can adaptively model the cross-modality partial relationship and further improve performances. Extensive experimental results show that our approach significantly improves the performances.

References

[1] Humam Alwassel, Dhruv Mahajan, Lorenzo Torresani, Bernard Ghanem, and Du Tran. Self-supervised learning by cross-modal audio-video clustering. In *NeurIPS*, 2019.

[2] Relja Arandjelovic and Andrew Zisserman. Look, listen and learn. In *ICCV*, 2017.

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

[4] Honglie Chen, Weidi Xie, Andrea Vedaldi, and Andrew Zisserman. Vggsound: A large-scale audio-visual dataset. In *ICASSP*, 2020.

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

[6] Chongjian Ge, Jiangliu Wang, Zhan Tong, Shoufa Chen, Yibing Song, and Ping Luo. Soft neighbors are positive supporters in contrastive visual representation learning. In *ICLR*, 2023.

[7] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, 2016.

[8] 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.

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

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

[11] Sangho Lee, Youngjae Yu, Gunhee Kim, Thomas M Breuel, Jan Kautz, and Yale Song. Parameter efficient multimodal transformers for video representation learning. In *ICLR*, 2021.

[12] Shuang Ma, Zhaoyang Zeng, Daniel McDuff, and Yale Song. Active contrastive learning of audio-visual video representations. In *ICLR*, 2020.

[13] Pedro Morgado, Ishan Misra, and Nuno Vasconcelos. Robust audio-visual instance discrimination. In *CVPR*, 2021.

[14] Pedro Morgado, Nuno Vasconcelos, and Ishan Misra. Audio-visual instance discrimination with cross-modal agreement. In *CVPR*, 2021.

[15] Aaron van den Oord, Yazhe Li, and Oriol Vinyals. Representation learning with contrastive predictive coding. *arXiv preprint arXiv:1807.03748*, 2018.

[16] Andrew Owens and Alexei A Efros. Audio-visual scene analysis with self-supervised multisensory features. In *ECCV*, 2018.

[17] Mandela Patrick, Yuki M Asano, Polina Kuznetsova, Ruth Fong, João F Henriques, Geoffrey Zweig, and Andrea Vedaldi. On compositions of transformations in contrastive self-supervised learning. In *ICCV*, 2021.

[18] 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.

[19] Du Tran, Heng Wang, Lorenzo Torresani, Jamie Ray, Yann LeCun, and Manohar Paluri. A closer look at spatiotemporal convolutions for action recognition. In *CVPR*, 2018.