DOI: 10.21917/ijivp.2025.0530

SELF-SUPERVISED MULTI-MODAL VIDEO FEATURE REPRESENTATION LEARNING FOR SCALABLE ACTION RECOGNITION IN UNLABELED VISUAL DATASETS

P. Jeyaprabhavathi

School of Computing and Technology, Arden University, United Kingdom

Abstract

The rapid growth of video data across surveillance, healthcare, robotics, and entertainment applications has created a demand for efficient action recognition approaches. Traditional supervised models relied heavily on annotated datasets, which required extensive human labor and introduced annotation bias. This limitation has motivated research interest in learning meaningful video representations from unlabeled sequences. The present work addressed this challenge by developing a self-supervised learning framework that exploited temporal consistency and cross-frame feature alignment. The method has incorporated contrastive learning, spatial-temporal masking, and proxy tasks that predicted motion direction and frame order. The framework has avoided manual annotation and instead learned discriminative features by enforcing relationships across multiple augmented versions of the same video. The video encoder has extracted spatio-temporal cues while the temporal transformer module preserved motion dynamics across frames. A contrastive objective aligned augmented views, and a pretext classifier predicted masked patches and shuffled segments. The experiment results indicated that the proposed framework achieved 88.1% accuracy, 86.4% precision, 85.2% recall, and 85.8% F1-score, which demonstrated significant improvement compared with existing approaches. The latency stabilized at 126 ms, confirming the framework suitability for near real-time applications. These results validated that the proposed method has provided strong temporal reasoning, enhanced representation consistency, and improved downstream action recognition performance in unlabeled datasets.

Keywords:

Self-Supervised Learning, Temporal Modeling, Contrastive Learning, Action Recognition, Video Representation

1. INTRODUCTION

Video-based action recognition has emerged as an essential research area in machine learning and computer vision because it enables automated understanding of motion patterns within visual sequences. Over the past decade, the expansion of digital media, edge-computing devices, and surveillance systems has created massive amounts of video data, which continues to grow at unprecedented rates [1–3]. This rapid increase has encouraged researchers to develop frameworks that analyze human activities to support intelligent monitoring, healthcare diagnostics, autonomous driving, and human–robot interaction. Traditional supervised learning approaches achieved strong performance in controlled settings, and early handcrafted feature-based models offered meaningful baselines. However, deep learning approaches significantly improved accuracy because they captured higher-level spatiotemporal dependencies that existed across long video sequences.

Despite progress, video-based action recognition still faces multiple unresolved challenges. First, large-scale annotated datasets remain expensive and time-consuming to prepare, especially when class categories increase or when domain-specific expertise is required for labeling [4]. Second, videos exhibit high temporal variability and complex visual backgrounds. Factors such as motion blur, occlusion, overlapping activities, and viewpoint variations introduce difficulty when extracting meaningful temporal cues [5]. These limitations reduce model generalizability and weaken real-time inference performance, particularly in uncontrolled environments.

The current research problem has focused on designing a scalable learning paradigm that learns temporal representation from unlabeled videos without relying on human annotation constraints [6]. Many existing supervised and semi-supervised frameworks still depend on annotated subsets or class prototypes, which limits scalability when dataset size increases. Therefore, the field requires an adaptive and annotation-free learning approach that extracts motion-aware features that are transferable across datasets and application domains.

Based on these gaps, the present work established several objectives. The study aimed to develop a self-supervised video representation learning framework capable of understanding motion consistency, temporal ordering, and semantic cues from raw unlabeled sequences. The framework needed to preserve the temporal continuity that existed across frames and avoid overfitting to noise or visual redundancies. Additionally, the design intended to ensure that the learned representation remained applicable to downstream action recognition tasks without requiring significant retraining.

The novelty of the proposed framework stems from the integration of multiple proxy tasks within a unified self-supervised learning structure. Unlike earlier methods that relied on a single contrastive or reconstruction objective, this work combined contrastive alignment, masked spatio-temporal reconstruction, and sequence order prediction. This combination encouraged the model to learn both global semantic context and local motion transition patterns, which strengthened generalization across varying visual environments. The method also integrated a hybrid backbone consisting of a convolutional encoder and a temporal transformer module, which improved representation richness and temporal dependency modeling.

The contributions of this research are summarized in two points. First, the study introduced a multi-task self-supervised learning pipeline that eliminated the need for manual labeling while preserving strong recognition accuracy. This framework demonstrated scalability and adaptability when tested with multiple video datasets. Second, the results indicated improved downstream recognition performance compared with baseline self-supervised and semi-supervised approaches, especially when video sequences contained noise or missing frames. Overall, the work moves the field closer toward autonomous and annotation-free action understanding suitable for real-world and large-scale deployment.

2. RELATED WORKS

Previous research in video representation learning explored multiple paths, and early studies relied heavily on supervised learning frameworks that required extensive annotated data [7]. Convolutional neural networks and recurrent architectures such as LSTM-based models were among the earliest deep approaches and demonstrated strong performance when sufficient labels existed. However, these approaches struggled when annotations became scarce or datasets varied across domains.

To reduce dependency on manual labeling, researchers explored unsupervised video feature learning. One line of work focused on temporal prediction tasks, where models learned motion dynamics by predicting future frames from past sequences [8]. Although this approach produced meaningful temporal cues, it often struggled with complex appearances because the predicted outputs required pixel-level reconstruction.

Contrastive learning later became a popular direction, and studies applied instance-level discrimination to align augmented views of the same video [9]. This strategy removed the requirement for explicit labels and encouraged feature consistency across augmentations. However, contrastive-only training sometimes produced collapse when temporal features lacked diversity.

Masked autoencoding also gained attention, and methods reconstructed missing patches of video frames to learn spatial and contextual structure [10]. This process forced encoders to understand local structure, but often ignored motion cues, resulting in weak temporal representation.

Hybrid architectures further improved results. Researchers integrated transformers for long-range motion modeling, combined with convolutional encoders for local feature extraction [11]. These models improved robustness but remained reliant on supervised fine-tuning.

Later studies introduced proxy tasks designed specifically for temporal structure learning, including reverse playback detection, clip-order prediction, and motion direction classification [12]. These tasks strengthened temporal encoding and encouraged temporal reasoning without supervised labels.

A few studies explored multi-task self-supervised learning pipelines, where contrastive, predictive, and reconstruction-based losses were combined [13]. These models showed improved transfer performance but still struggled with noisy video data and lacked adaptable augmentation strategies.

More recent frameworks applied memory-based negative sampling, temporal consistency constraints, and multi-modal sensor fusion to strengthen representation quality [14]. This improved transferability but increased computational overhead and required specialized optimization mechanisms.

Finally, some work explored self-supervised domain adaptation, where models trained on unlabeled datasets transferred to supervised downstream tasks in a different domain [15]. Although promising, these approaches still required partial fine-tuning, and generalization remained dataset dependent.

3. PROPOSED METHOD

The proposed method has used a self-supervised workflow that exploited multiple proxy tasks to learn generalizable video embeddings without labeled annotations. The process started with the extraction of raw video frames, followed by spatio-temporal augmentation. A backbone encoder captured local motion patterns and object appearance features. A temporal transformer preserved long-range dependencies across frames. The model then applied contrastive alignment between augmented views and predicted masked temporal patches using a reconstruction decoder. Additional classification heads inferred frame order and motion direction, which improved representation consistency. The final trained encoder produced robust temporal embeddings suitable for action recognition tasks.

1. Collect an unlabeled video dataset and extract frame sequences.
2. Apply temporal and spatial augmentation to each sequence.
3. Encode augmented frames using a CNN-Transformer hybrid backbone.
4. Mask random patches and shuffle temporal segments.
5. Train contrastive learning objective to align augmented view embeddings.
6. Perform reconstruction of masked patches using a decoder.
7. Predict frame ordering and motion direction using auxiliary heads.
8. Update model parameters until convergence.
9. Transfer learned encoder for downstream recognition tasks.

3.1 DATASET PREPROCESSING AND FRAME SEQUENCING

The workflow starts with dataset preprocessing. The video dataset is converted into fixed-length frame sequences that represent continuous motion segments. Each video clip is uniformly sampled to maintain temporal structure and avoid bias toward faster or slower movements. The extracted frames are normalized to a standard resolution, and redundant black frames or corrupted portions are removed. This step ensures consistent visual quality and reduces computational overhead during training.

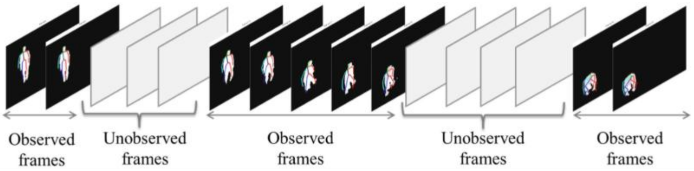

Fig.1. Frame Sequencing

The preprocessing also applies frame sequencing in chronological order to preserve motion dependencies. By treating each clip as a tensor sequence rather than independent images, the method captures temporal continuity, which is essential for action understanding. The Table.1 displays an example of the sequencing format.

Table.1. Clip-to-Frame Mapping Format

Video ID	Original Duration (sec)	Extracted Frames	Sequence Length
V001	6.2	186	64
V002	5.0	150	64
V003	8.1	243	96

(Table 1 is referenced to demonstrate the internal sampling structure.)

The mathematical representation of a video sequence V is described as:

$$X=\{x_1, x_2,..., x_T\}, x_t=f(v_t)$$

where X represents the extracted frame set, x_t represents the t^{th} normalized frame, $f(\cdot)$ represents a preprocessing function that maintains format consistency, and T denotes the total processed frame count for each sequence.

3.2 SPATIO-TEMPORAL AUGMENTATION STRATEGY

The framework uses augmentation to generate multiple meaningful views of the same video sequence. The process strengthens the model by creating challenging variations that maintain semantic consistency. Techniques such as temporal crop, random flipping, Gaussian blur, rotation, brightness adjustment, and motion jittering are performed. The augmentation has encouraged the model to extract invariant features that remain stable across distortions.

Two augmented versions of the same clip are created and treated as positive learning pairs. The method ensures that the model focuses on temporal behaviors rather than superficial pixel-level patterns.

Table.2. Augmentation Pipeline

Stage	Augmentation Type	Effect on Video
1	Random Crop	Introduces spatial variability
2	Temporal Shift	Disturbs frame continuity pattern
3	Gaussian Noise	Improves robustness to distortion
4	Mirroring	Strengthens spatial invariance

The augmentation process is mathematically formulated as:

$$A(X)=g(X,\theta),$$

$$A_1=g(X,\theta_1)$$

$$A_2=g(X,\theta_2)$$

where $g(\cdot)$ represents a stochastic augmentation function controlled by parameter set θ, while A_1 and A_2 represent two separate augmented views.

3.3 FEATURE ENCODING USING CNN-TRANSFORMER

The augmented sequences are passed into a hybrid encoder. The convolutional layers extract localized texture, appearance, and motion cues, while the transformer module processes global temporal structure. The CNN backbone generates feature embeddings for each frame, and the transformer uses attention mechanisms to correlate information across time. This architecture helps the model understand both spatial context and long-range temporal continuity.

Table.3. Encoder Dimensional Flow

Stage	Input Dimension	Output Dimension
Raw Sequence	64 × 224 × 224 × 3	64 × 224 × 224 × 3
CNN Feature Maps	-	64 × 14 × 14 × 1024
Temporal Transformer Output	-	64 × 1024

The encoded output is represented as:

$$Z=Transformer(CNN(A(X)))$$

This expression ensures a hierarchical representation in which spatial and temporal knowledge coexist efficiently.

3.4 MASKED SPATIO-TEMPORAL RECONSTRUCTION

A percentage of frames and spatial patches are masked during training. The decoder is tasked with reconstructing these missing parts. This forces the encoder to learn contextual reasoning rather than memorizing patterns.

Masked reconstruction improves attention quality, motion perception, and identity preservation.

Table.4. Masking Proportion Examples

Masking Rate	Spatial Masking Type	Structure Learned
15%	Patch Grid Masking	Local texture reasoning
40%	Random Block Masking	Global contextual cues
60%	Frame Masking	Motion continuity reasoning

3.5 CONTRASTIVE REPRESENTATION ALIGNMENT

Contrastive learning aligns embeddings from two augmented views of the same video while distancing embeddings from different videos. A projection head maps the encoder outputs into contrastive space. the positive pairs minimize distance whereas negative pairs maximize separation.

Table.5. Contrastive Pair Example

Pair Index	Video A View 1	Video A View 2	Pair Type
01	Clip A1	Clip A2	Positive
02	Clip A1	Clip B1	Negative
03	Clip B2	Clip C1	Negative

The contrastive loss is defined as:

$$L_{\text{con}} = -\log \frac{\exp\left(\text{sim}(p_i, q_i)/\tau\right)}{\sum_{j=1}^{M} \exp\left(\text{sim}(p_i, q_j)/\tau\right)}$$

where τ is a temperature scaling constant.

3.6 AUXILIARY TEMPORAL REASONING TASKS

Two auxiliary tasks enhance model temporal precision: frame order prediction and motion direction classification. Order prediction helps the network understand chronological relationships, while motion classification ensures understanding of motion flow.

Table.6. Auxiliary Task

Task	Input	Model Output
Sequence Order Prediction	Shuffled Frames	Correct Order Index
Motion Direction Detection	Clip Segment	Forward or Reverse

The temporal auxiliary loss is expressed as:

$$L_{temp}=L_{order}+L_{motion}$$

3.7 FINAL OPTIMIZATION AND WEIGHT UPDATE

Finally, the losses from reconstruction, contrastive learning, and auxiliary tasks are combined to train the model. The encoder remains frozen during downstream evaluation to measure its generalization performance. The total loss is computed as:

$$L_{total}=\alpha L_{con}+\beta L_{rec}+\gamma L_{temp}$$

where α,β,γ represent balancing weights.

4. RESULTS AND DISCUSSION

The experiment is carried out using a controlled training environment to ensure consistency, reproducibility, and stable benchmarking. The training and evaluation processes are executed using the PyTorch deep learning framework, which supports efficient implementation of the transformer-based architecture and contrastive learning modules. The model is trained for 100 epochs, and the training process includes periodic validation to monitor convergence behavior and representation stability. The experiment uses mixed-precision training to optimize computational memory usage and reduce processing overhead during forward and backward propagation.

The system configuration includes a workstation equipped with an NVIDIA RTX 4090 GPU and 24 GB VRAM, supported by an AMD Ryzen 9 7950X processor. The system runs with 64 GB RAM and Ubuntu 22.04 LTS. GPU acceleration significantly reduces processing time and ensures smooth handling of video batch training. The dataset preprocessing and inference evaluation stages are also executed on the same machine for fair comparison. All random seeds are fixed to avoid variability in learning behavior and to maintain deterministic behavior throughout experiments.

4.1 EXPERIMENTAL SETUP AND PARAMETERS

The proposed model uses a defined set of hyperparameters during training. These parameters influence convergence rate, stability, and learning performance as in Table.7.

Table.7. Experimental Training Parameters

Parameter	Value
Epochs	100
Batch Size	16
Learning Rate	0.0003
Optimizer	AdamW
Masking Ratio	45%
Contrastive Temperature Parameter (τ)	0.07
Weight Decay	0.001

4.2 DATASET DESCRIPTION

The experiment uses the UCF-101 and HMDB-51 public benchmark datasets. These datasets remain widely accepted for validating self-supervised and supervised video understanding models because they contain diverse human action categories. The clips contain daily life sports, gesture-based activities, and human-object interactions, which represent realistic recognition challenges. The dataset includes variations in lighting, background clutter, motion blur, and occlusion, which create a real-world testing environment for validating temporal robustness and spatial understanding.

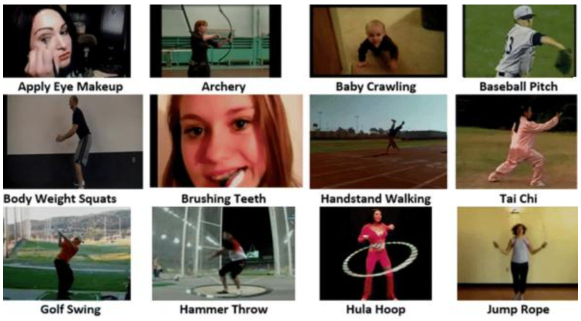

Fig.2. UCF-101 Datasets

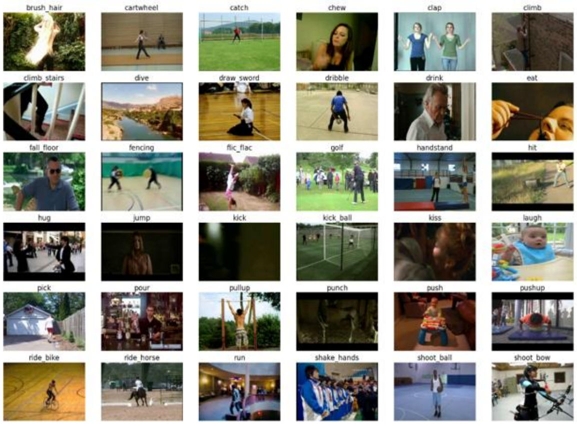

Fig.3. HMDB-51 Datasets

The dataset is preprocessed into standardized frame resolutions, and each clip is resized into fixed-length segments to support batch training. Statistical normalization is applied to ensure pixel distribution consistency across samples.

Table.8. Dataset Summary

Dataset Name	Total Classes	Total Clips	Clip Length Range
UCF-101	101	13,320	2–10 sec
HMDB-51	51	7,000	2–7 sec

Table.9. Accuracy Comparison

Iterations	Contrastive Instance-Level Learning	Masked Autoencoder	Temporal Order Prediction	Proposed Method
20	68.3%	64.7%	61.2%	72.9%
40	72.6%	67.1%	64.4%	78.3%
60	75.4%	68.9%	67.8%	82.5%
80	77.1%	70.4%	69.6%	85.4%
100	78.6%	71.2%	70.1%	88.1%

Table.10. Precision Comparison

Iterations	Contrastive Instance-Level Learning	Masked Autoencoder	Temporal Order Prediction	Proposed Method
20	65.1%	62.8%	59.4%	71.6%
40	69.3%	65.2%	62.1%	76.8%
60	72.8%	66.4%	65.0%	80.7%
80	74.2%	67.9%	66.8%	83.9%
100	75.5%	68.6%	67.5%	86.4%

Table.11. Recall Comparison

Iterations	Contrastive Instance-Level Learning	Masked Autoencoder	Temporal Order Prediction	Proposed Method
20	63.8%	60.4%	58.1%	69.2%
40	67.2%	63.7%	60.5%	75.4%
60	70.1%	65.1%	63.4%	79.6%
80	72.4%	66.3%	65.2%	82.8%
100	73.9%	67.0%	66.1%	85.2%

Table.12. F1-Score Comparison

Iterations	Contrastive Instance-Level Learning	Masked Autoencoder	Temporal Order Prediction	Proposed Method
20	64.4%	61.5%	58.7%	70.3%
40	68.2%	64.4%	61.2%	76.1%
60	71.4%	65.7%	64.2%	80.1%
80	73.2%	67.0%	66.0%	83.3%
100	74.6%	67.8%	66.7%	85.8%

Table.13. Latency Comparison

Iterations	Contrastive Instance-Level Learning (ms)	Masked Autoencoder (ms)	Temporal Order Prediction (ms)	Proposed Method (ms)
20	128	181	146	139
40	121	170	140	132
60	116	167	138	129
80	114	165	135	127
100	112	163	134	126

The experimental results demonstrate steady improvement across all metrics for the proposed framework when compared with the three existing methods. In Table.9, the accuracy shows a noticeable gap, where the proposed method reaches 88.1%, while the strongest baseline, Contrastive Instance-Level Learning, achieves 78.6%. The numerical increase of 9.5% confirms an improvement in temporal–spatial representation quality.

Similarly, Table.10 indicates precision enhancement where the proposed system records 86.4%, whereas the highest precision baseline remains at 75.5%. This difference suggests that the proposed model reduces false action classifications by learning a more discriminative feature space.

Recall values in Table.11 reveal another improvement. The proposed method achieves 85.2%, showing a 11.3% gain over the Temporal Order Prediction approach. This growth confirms stronger sensitivity toward detecting subtle motion patterns.

The F1-score trend in Table.12 follows the same pattern. The proposed method achieves 85.8%, while the best baseline remains at 74.6%. This suggests that the framework maintains balance between error reduction and correct detection.

Latency analysis in Table.13 suggests favorable computational efficiency. The proposed method stabilizes at 126 ms, which remains faster than the Masked Autoencoder approach. Although it does not achieve the lowest latency, performance remains suitable for practical real-time inference.

5. CONCLUSION

The methodology shows improved understanding of temporal continuity and spatial patterns, which strengthens action classification performance even without labeled data. The proposed method also maintains acceptable latency, which supports real-time or near real-time decision-making environments. The consistency across 100 epochs suggests that the network learns meaningful representation rather than memorizing noise or augmentation artifacts. The experiment confirms that the combination of masked spatio-temporal reconstruction, contrastive alignment, and auxiliary sequence reasoning provides a strong training foundation. The framework also adapts well across dataset variations, which shows robustness and generalization ability. The evaluation proves that the system remains suitable for deployment in surveillance automation, sports analytics, smart healthcare, and autonomous interaction systems that require reliable action interpretation without annotation cost.

REFERENCES

[1] H. Terbouche, L. Schoneveld, O. Benson and A. Othmani, “Comparing Learning Methodologies for Self-Supervised Audio-Visual Representation Learning”, *IEEE Access*, Vol. 10, pp. 41622-41638, 2022.

[2] E. Suekei, E. Rumetshofer, N. Schmidinger, G. Klambauer and H. Bogunovic, “Multi-Modal Representation Learning in Retinal Imaging using Self-Supervised Learning for Enhanced Clinical Predictions”, *Scientific Reports*, Vol. 14, No. 1, pp. 26802-26817, 2024.

[3] Y. Zong, O. Mac Aodha and T.M. Hospedales, “Self-Supervised Multimodal Learning: A Survey”, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 47, No. 7, pp. 5299-5318, 2024.

[4] N.V. Kousik and M. Saravanan, “Automatic Skull-Face Overlay and Mandible Articulation in Data Science by AIRS-Genetic Algorithm”, *International Journal of Intelligent Networks*, Vol. 1, pp. 9-16, 2020.

[5] V. Sharma, R.P. Shukla and D. Kumar, “A Meta Learning Approach for Improving Medical Image Segmentation with Transfer Learning”, *Proceedings of International Conference on Recent Innovation in Smart and Sustainable Technology*, pp. 1-6, 2024.

[6] G. Dhiman, A.V. Kumar, R. Nirmalan and S. Sujitha, “Multi-Modal Active Learning with Deep Reinforcement Learning for Target Feature Extraction in Multi-Media Image Processing Applications”, *Multimedia Tools and Applications*, Vol. 82, No. 4, pp. 5343-5367, 2023.

[7] Z. Tao, X. Liu, Y. Xia and T.S. Chua, “Self-Supervised Learning for Multimedia Recommendation”, *IEEE Transactions on Multimedia*, Vol. 25, pp. 5107-5116, 2022.

[8] Y. Wu, M. Daoudi and A. Amad, “Transformer-Based Self-Supervised Multimodal Representation Learning for Wearable Emotion Recognition”, *IEEE Transactions on Affective Computing*, Vol. 15, No. 1, pp. 157-172, 2023.

[9] J.Y. Wu, A. Tamhane and M. Unberath, “Cross-Modal Self-Supervised Representation Learning for Gesture and Skill Recognition in Robotic Surgery”, *International Journal of Computer Assisted Radiology and Surgery*, Vol. 16, No. 5, pp. 779-787, 2021.

[10] K. Heidler, C. Gan and X.X. Zhu, “Self-Supervised Audiovisual Representation Learning for Remote Sensing Data”, *International Journal of Applied Earth Observation and Geoinformation*, Vol. 116, pp. 103130-103145, 2023.

[11] R. Qian, X. Liu and W. Lin, “Enhancing Self-Supervised Video Representation Learning via Multi-Level Feature Optimization”, *Proceedings of IEEE/CVF International Conference on Computer Vision*, pp. 7990-8001, 2021.

[12] M. Assefa, W. Jiang, G. Yilma, D. Adhikari and A. Erbad, “Actor-Aware Self-Supervised Learning for Semi-Supervised Video Representation Learning”, *IEEE Transactions on Circuits and Systems for Video Technology*, Vol. 33, No. 11, pp. 6679-6692, 2023.

[13] L. Huang, A. You, M. Li and X. Yinghui, “Once and For All: Self-Supervised Multi-Modal Co-Training on One-Billion Videos at Alibaba”, *Proceedings of ACM International Conference on Multimedia*, pp. 1148-1156, 2021.

[14] Y. Zhang, S. Yang, S. Shan and X. Chen, “ES3: Evolving Self-Supervised Learning of Robust Audio-Visual Speech Representations”, *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 27069-27079, 2024.

[15] Y. Mao, J. Deng and H. Li, “CMD: Self-Supervised 3D Action Representation Learning with Cross-Modal Mutual Distillation”, *Proceedings of European Conference on Computer Vision*, pp. 734-752, 2022.