Temporal Relations of Informative Frames in Action Recognition

Alireza Rahnama
Department of Engineering
Faculty of Electrical and Computer Engineering
Kharazmi University
Tehran, Iran
Email: Alireza.rahnama@khu.ac.ir

Azadeh Mansouri
Department of Engineering
Faculty of Electrical and Computer Engineering
Kharazmi University
Tehran, Iran
Email: A_mansouri@khu.ac.ir

***Abstract*—This paper presents a simple approach leveraging temporal learning on informative frames for action recognition. We propose a training-free simple adaptive frame selection scenario employing just the similarity technique in a temporal window. The proposed frame selection method provides an appropriate strategy to capture informative frames and provide meaningful features. Moreover, we use transfer learning for spatial feature extraction and employ LSTM and GRU for temporal modeling. Our method is evaluated on two popular datasets, UCF11 and KTH, and it demonstrates acceptable results.**

***Keywords*—Action recognition, Frame selection, deep temporal modeling, transfer learning, Spatial-Temporal features**

I. Introduction

The proliferation of digital devices has resulted in an enormous amount of video data being collected daily. As a result, understanding and analyzing this video data has become one of the primary tasks of computer vision experts. Identifying human activities has been widely used in vision tasks such as violence detection [1]–[3], monitoring people's behavior in public places [4], [5], and examining the behavior of the elderly [6], [7].

Video processing is a computationally expensive task. Most efforts have been focused on accuracy improvements by devising larger architectures. Yet, designing a low-complexity system can provide more benefits, especially in real-time applications. Deep learning techniques make use of 2D-CNNs [8]–[10], 3D-CNNs [11], [12], or both [13] for action recognition. 3D CNN demonstrated acceptable results using 3D convolution leads to high computational cost. In many applicable scenarios, spatial features are extracted using 2D CNN and then temporal pooling is employed to illustrate relations over time.

Depending on the subject matter or other factors, a video may have more informative sections than others. [14] provides experiments demonstrating that selecting the ideal number of frames provides more accurate classification results. In another experiment, [15] demonstrates that some informative frames provide sufficient discrimination for a human visual system (HVS). This implies the innate redundancy in the video signals.

In this paper, we utilized an adaptive frame selection and employed different methods for temporal pooling to present the applicable method; To extract the spatial frame-based feature, we used ResNet-50 [16] to obtain spatial features; Temporal features were obtained using the LSTM network, and subsequently, a fully connected network was trained to classify these features.

II. Related work

Several methods, including CNNs, time sequence models, and hybrid approaches, have been employed to identify activities. In fact, videos have a lot of space and time redundancy, which is one of the fundamental challenges. For video classification tasks, analyzing both spatial and temporal features seems to be necessary. Traditional methods have generated descriptors with visual characteristics using optical flow [17], MBH [8], and artificial features [18].

2D CNN methods utilize each frame as the input of convolutional models to obtain spatial features. A new CNN model was built [17] for the training stage, whereas some articles [19] [20] extracted feature vectors using pre-trained models such as VGG16, ResNet-50 and AlexNet. 2D CNN feature extractor provides spatial frame-based features. Yet, their correlation is essential for action recognition. While several articles employed 3d CNN to extract spatial and temporal features [21] and [22], this approach has some disadvantages. For instance, they require greater computational resources than RNNs since 3D CNNs compute convolutions over both spatial and temporal dimensions, meanwhile, RNNs are better at capturing long-term temporal relations.

To extract temporal features, deep temporal modeling could be used such as [19] and [23]. A video may have more discriminating frames than others. In [14] experimental results provided to this intuition. Selecting the informative number of frames can provide more accurate results than employing the full movie. Many methods are presented to reduce the computational complexity [24], [25].In these methods,

979-8-3503-5049-4/24/$31.00

simple techniques are used to identify the most significant and informative regions for action recognition. However, these methods are learning-based and need to be trained especially for long videos. In this paper, we introduce a training-free simple adaptive frame selection employing just the similarity technique in a temporal window. Then for the video-based action recognition, the temporal aggregation method is applied to informative selected frames. The rest of the paper is as follows: in section III the proposed method is described in detail. The experimental results are given in section IV and the conclusion is explained in section V.

III. PROPOSED METHOD

The main architecture of the proposed method is similar to CNN-LSTM. In fact, frame-based features are temporally pooled using different polling strategies to provide video-based features. Furthermore, a simple scheme for informative frame selection during the temporal window is employed. Figure 1 shows the overall structure of the system at various stages. The frame selection algorithm is the first part of our framework and it helps to select informative and distinct frames as input. Then, the pre-trained model feeds by all of the selected frames. The video's features are extracted using a pre-trained ResNet-50, and the result of each frame is a feature vector of size 1×2048. After frame-based feature extraction, two scenarios for feature vector selection are used; the first includes dividing each video into K-frame segments, and the second is the utilization of Max-pooling to have efficient feature vectors. Mostly, the Max-pooling strategy is employed for the evaluation of the clips and K-frame segments are utilized for video evaluation with longer length. Then, LSTM and GRU as the Deep temporal models are employed to process each feature vector that has been obtained from the previous step to provide the video-based feature. The output stage is dense layers of nodes and the final layer has a node for each of the classes. In the following subsection, these steps are described.

A. Frame selection algorithm

There is a high degree of similarity between consecutive video frames. For instance, whenever the camera is stationary, a large fraction of the recorded frames are just the background of the location as captured by the camera; little movement is taking place in that zone. With the help of informative frame selection, duplicate and similar objects can be avoided for detection and processing; only distinctive frames can be chosen as the input. This approach selects the first frame because it has an extensive amount of information about the video content, including background and object details; Next, we use formula 1 [26] to determine the amount of similarity between the first and second frames. This provides the similarity matrix, and we

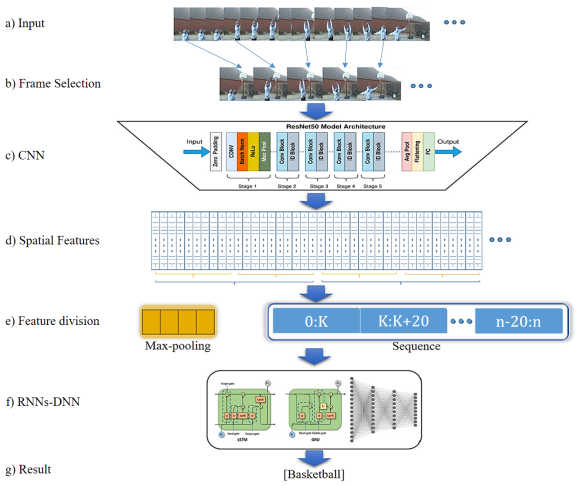

Fig. 1. This figure illustrates all components of our framework. (a)Input video b)Selected frames (c) ResNet-50 (d) Spatial features are extracted by pre-trained model (e)Feature vector selection (f)Combination between LSTM or GRU with neural network (g) Result

average all of the elements of this matrix to have a scalar number as the similarity between these two frames. The following formula represents the similarity frame statement, where $F(i)$ is the current frame and $SF(i)$ is the previously selected frame.

$$Frame - Similarity = \frac{2 \times F(i) \times SF(i)}{F(i)^2 + SF(i)^2} \quad (1)$$

the scalar similarity is $FM(i)$ for each selected frame and we keep it in an array. The next frame is chosen by comparing the last frame that was selected with the new frame using the formula 1. Afterward, we compare the similarity of the two frames to the similarity of the selected frames. If the similarity is less than the amount of the previous similarities, it indicates that the new frame is distinctive and includes important information; if it is equal to or greater than the amount of the previous similarities, it has not been selected. By calculating the average of all the values in the Mean window list in formula 2, we can compare each new frame to the average of all the previously selected frames. In the proposed method 10-frame window is determined to ensure a better assessment, the new frame will be chosen if it differs from the 10 previously chosen frames [27]. In fact, the threshold is the average similarity value between the last 10 selected frames, we compare the new frame with this threshold.

$$Mean - window = \frac{\sum_{i=0}^{n} FM(i)}{n} \quad (2)$$

B. Spatial features

We can tackle a different problem by using a pre-trained model with additional data. For a few reasons, including the outcomes of our previous research, FLOPs, versatility, accuracy, and residual blocks and skip connections, we utilized ResNet-50 for feature

extraction. Each selected frame feeds to the ResNet-50 to extract features. Several convolutional layers in these models apply various filters to the input image, producing a set of feature maps. From simple shapes and textures in the upper levels to more intricate and abstract concepts in the lower layers, the feature maps illustrate the spatial hierarchy of features in the frame. The next stage involves choosing the feature vectors. Our paper suggests two techniques. The initial technique employs Max-pooling to provide powerful feature vectors as temporal network input, which allows to reduce testing times and create a less complicated network. as in:

- Max-Pooling: temporal pooling data augmentation on each video allows us to have more information, we split up each video's vectors into 4 portions, and after that, our pooled time series (PoT) is constructed by applying the Max-pooling operator as illustrated in formula 3 over each temporal portion of video to provide a powerful feature vector for each part [10]. The advantage of this scenario is we can employ more straightforward networks and save time due to having only four powerful spatiotemporal feature vectors for each video.

$$X^{max}(t^s, t^e) = Maxf(i(t)), \quad t = t_s, \ldots, t_e \tag{3}$$

- K-frame segment: In this case, we divided the feature vectors into K sections. After that, every video is split into equal segments of K frames and each of these parts has its video label. The formula 4 is used to select each part of the feature vectors, where f represents the feature vector and i represents the frame number. After this step, the feature vectors without using Max-pooling are used as input for the Temporal models. As mentioned before, this approach is appropriate for long-length and high-frame-count videos, as the main goal is to augment the data for the training phase.

$$Video-Seq = \sum_{i=1}^{n} f[(20i-19) : (20i+1)] \tag{4}$$

C. Temporal modeling

To thoroughly examine each method, we employed three popular techniques for extracting temporal features, which have all been applied in numerous articles. We have provided each of these models with the input data so that they can extract the time-related information of feature vectors using a memory and filter. In the end, a fully connected network is used to classify these collected spatio-temporal characteristics.

- LSTM: Long short-term memory (LSTM) [28] has been employed to obtain the temporal characteristics and the relations of the feature vectors and, consequently, have a decent classification based on the memory that these models contain.
- GRU: we utilized the Gating Recurrent Unit (GRU) [29] Due to its more uncomplicated structure, which is comparable to LSTM. GRUs were introduced as a good substitute to capture dependencies for sequences with time intervals since, like LSTMs, they do not suffer from the issue of vanishing gradients of standard RNNs. They work well and are comparable to LSTMs, as indicated in Table 1, and have fewer indicators, less complexity, and a quicker training time. Similar to LSTM models as explained in section IV, the GRU model's configuration is determined based on the amount of input data and feature selection scenarios for the UCF11 and KTH datasets.

IV. EXPERIMENTAL RESULTS

We evaluate and classify the UCF11 and KTH datasets. As previously mentioned, ResNet-50 is selected as a pre-trained model to extract spatial features through selected frames. After that, well-known temporal sequence modeling techniques including LSTM, and GRU are applied to the classification of this divided or pooled spatial information. The number of LSTM units is changed depending on the various scenarios and different numbers of input data from 40 to 80. We also selected different batch sizes such as 64, 128, and 256. The optimizer function for some training phases with a large number of epochs was SGD and for a short number of epochs was Adam, which allowed us to reach the desired result after 60 to 300 epochs. For instance, for all results on the KTH dataset in Table II we used 80 units of GRU and LSTM with different batch sizes and epochs.

A. Experimental Results of Frame Selection

We will begin by looking through the positive aspects of our frame selection method in this section. As shown in Table I, The entire number of frames in the datasets is displayed in the first column, which indicates that all of these frames ought to be processed, trained on, and tested; The second column illustrates the number of selected frames by the proposed frame selection method; The last column provides the proportion of frames that have been performed. It is clearly shown that activity recognition is performed using around 30% of the data. This demonstrates the efficiency of the algorithm, which allows us to reduce considerable amounts of time and memory by avoiding redundancies.

TABLE I
COMPARISON OF TWO SCENARIOS: WITH OR WITHOUT FRAME SELECTION

Dataset	***All frames***	***selected frames***	***Selected frame's percentage***
KTH	288,906	83,280	28.5%
UCF11	304,235	95,108	32.8%

B. Evaluation on KTH dataset

One of the most frequently utilized public datasets for human action is the KTH. The six human activity categories in this collection—walking, running, slow running, boxing, waving hands, and clapping hands are carried out repeatedly by 25 participants in four different settings, one of which is a space, open s1, outside with a change home s2, outside with new clothes s3, and indoors or in a closed environment s4. This dataset contains 599 videos and all of them were taken with a fixed camera at a frame rate of 25 frames on homogeneous terrains with a resolution of 120×160 pixels. We employed the standard method for evaluating our framework on the KTH dataset. This included having 25 participants act the videos for each action, videos of 16 actors utilized for training and validation, and videos related to the remaining 9 actors [2, 3, 5, 6, 7, 8, 9, 10, 22] being considered for testing [30]. We then used ResNet-50 to extract spatial features of selected frames and tested the LSTM and GRU networks using two separate scenarios maxpooling and K-feature vector segment to obtain additional information. The Table II summarizes these levels of accuracy using different methods. In all the experiments, K is empirically selected as 20.

TABLE II
PERFORMANCE OF THE PROPOSED SCENARIOS ON THE KTH DATASET

Method	***Result***
MP + LSTM	94.44%
MP + GRU	92.59 %
Kseq + LSTM	94.45%
Kseq + GRU	**95.38%**

In Table III the performance results of the proposed method and the other methods are depicted. It clearly shows the effectiveness of our simple proposed approach. The results are Superior and in the case of [31] which utilized dense trajectories from consecutive frames, our results are comparable.

TABLE III
PERFORMANCE COMPARISON OF THE PROPOSED APPROACH WITH THE STATE-OF-THE-ART ON THE KTH DATASET

Method	***Year***	***Result***
3d CNN [21]	2012	90.2%
CNN-LSTM [23]	2020	93.86%
Differential RNN [32]	2015	93.96%
Dense trajectory [33]	2016	94.2%
3D-ConvNet + LSTM [22]	2011	94.39%
SIFT+OF+CNN [17]	2020	94.96%
DTD, DNN [31]	2015	**95.6%**
Our method (Kseq + GRU)	-	95.38%

C. Evaluation on UCF11 dataset

Due to its numerous variations in camera movement, lighting, viewing angle, cluttered background, etc., the UCF11 dataset is more challenging. 1,600 videos constitute the collection, and each one is categorized into one of the following 11 action sports: volleyball, trampoline, basketball, horseback riding, basketball, cycling, diving, golf, and swing [34]. The Kseq feature segmentation is only applied to the KTH since the UCF11 dataset has separated each video into smaller portions (clips). In short videos like UCF11 dataset videos, LSTM and GRU results applied on the Max-pooled data which are illustrated in TableIV.

TABLE IV
PERFORMANCE OF THE PROPOSED SCENARIOS ON UCF11 DATASET

Method	***LOOCV***	***5FoldCV***	***10FoldCV***
MP + LSTM	**98.27%**	**97.18%**	**97.69%**
MP + GRU	97.63%	96.99%	97.63%

The performance of the proposed approach is evaluated using three scenarios: LOOCV, 5foldCV, and 10foldCV in which the results are illustrated in Table IV. In Table V the best-proposed results in the Leave-One-Out-Cross-Validation standard are compared with existing methods. The results are comparable with [35] and better than the other methods. It should be noticed that [35] utilizes the actor's position in each frame and estimates the body's motion for feature illustration. Then, a time series approach is employed for classification.

TABLE V
PERFORMANCE COMPARISON OF THE PROPOSED APPROACH WITH THE STATE-OF-THE-ART ON UCF11 DATASET

Method	***LOOCV***
Dense Trajectories [33]	84.2%
Local Motion [36]	86.1%
KMP [18]	87.6%
Gaurav Yadav [37]	91.3%
Two Stream LSTM [19]	94.6%
ViCo-MoCo-DL [38]	97%
Pose Descriptor [35]	**99.1%**
Our method(MP + LSTM)	98.27%

V. CONCLUSION

In this paper, we introduce a training-free approach that utilizes an informative-frame selection algorithm, proving particularly effective for videos with extensive data. For frame-level feature extraction, we employ the pre-trained ResNet-50. In the temporal modeling segment, we assess the accuracy of video-sequence and Max-pooling scenarios when coupled with LSTM and GRU. The framework is evaluated on two publicly available datasets, KTH and UCF11, demonstrating its ability to achieve state-of-the-art results. Future

research endeavors include leveraging larger datasets and analyzing the effectiveness of incorporating the transformer model in the final section of the framework.

REFERENCES

[1] N. Honarjoo, A. Abdari, and A. Mansouri, "Violence detection using pre-trained models," in *2021 5th International Conference on Pattern Recognition and Image Analysis (IPRIA)*, pp. 1–4, IEEE, 2021.

[2] N. Honarjoo, A. Abdari, and A. Mansouri, "Violence detection using one-dimensional convolutional networks," in *2021 12th International Conference on Information and Knowledge Technology (IKT)*, pp. 188–191, IEEE, 2021.

[3] F. U. M. Ullah, M. S. Obaidat, A. Ullah, K. Muhammad, M. Hijji, and S. W. Baik, "A comprehensive review on vision-based violence detection in surveillance videos," *ACM Computing Surveys*, vol. 55, no. 10, pp. 1–44, 2023.

[4] A. S. Patel, R. Vyas, O. Vyas, M. Ojha, and V. Tiwari, "Motion-compensated online object tracking for activity detection and crowd behavior analysis," *The Visual Computer*, vol. 39, no. 5, pp. 2127–2147, 2023.

[5] M. Salvatori, V. Oberosler, M. Rinaldi, A. Franceschini, S. Truschi, P. Pedrini, and F. Rovero, "Crowded mountains: Long-term effects of human outdoor recreation on a community of wild mammals monitored with systematic camera trapping," *Ambio*, vol. 52, no. 6, pp. 1085–1097, 2023.

[6] K. Durga Bhavani and M. Ferni Ukrit, "Design of inception with deep convolutional neural network based fall detection and classification model," *Multimedia Tools and Applications*, pp. 1–19, 2023.

[7] M. Darvish and A. Mansouri, "Compressed domain human fall detection using deep features," 2023.

[8] L. Wang, Y. Xiong, Z. Wang, Y. Qiao, D. Lin, X. Tang, and L. Van Gool, "Temporal segment networks: Towards good practices for deep action recognition," in *European conference on computer vision*, pp. 20–36, Springer, 2016.

[9] A. Abdari, P. Amirjan, and A. Mansouri, "Speeding up action recognition using dynamic accumulation of residuals in compressed domain," *arXiv preprint arXiv:2209.14757*, 2022.

[10] M. S. Ryoo, B. Rothrock, and L. Matthies, "Pooled motion features for first-person videos," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 896–904, 2015.

[11] K. Hara, H. Kataoka, and Y. Satoh, "Can spatiotemporal 3d cnns retrace the history of 2d cnns and imagenet?," in *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, pp. 6546–6555, 2018.

[12] D. Tran, L. Bourdev, R. Fergus, L. Torresani, and M. Paluri, "Learning spatiotemporal features with 3d convolutional networks," in *Proceedings of the IEEE international conference on computer vision*, pp. 4489–4497, 2015.

[13] C. Luo and A. L. Yuille, "Grouped spatial-temporal aggregation for efficient action recognition," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 5512–5521, 2019.

[14] D.-A. Huang, V. Ramanathan, D. Mahajan, L. Torresani, M. Paluri, L. Fei-Fei, and J. C. Niebles, "What makes a video a video: Analyzing temporal information in video understanding models and datasets," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 7366–7375, 2018.

[15] L. Sevilla-Lara, Y. Liao, F. Güney, V. Jampani, A. Geiger, and M. J. Black, "On the integration of optical flow and action recognition," in *Pattern Recognition: 40th German Conference, GCPR 2018, Stuttgart, Germany, October 9-12, 2018, Proceedings 40*, pp. 281–297, Springer, 2019.

[16] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 770–778, 2016.

[17] J. Basavaiah and C. G. Patil, "Human activity detection and action recognition in videos using convolutional neural networks," *Journal of Information and Communication Technology*, 2020.

[18] L. Shao, L. Liu, and M. Yu, "Kernelized multiview projection for robust action recognition," *International Journal of Computer Vision*, vol. 118, pp. 115–129, 2016.

[19] H. Gammulle, S. Denman, S. Sridharan, and C. Fookes, "Two stream lstm: A deep fusion framework for human action recognition," in *2017 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pp. 177–186, 2017.

[20] M. Ravanbakhsh, H. Mousavi, M. Rastegari, V. Murino, and L. S. Davis, "Action recognition with image based cnn features," *ArXiv*, vol. abs/1512.03980, 2015.

[21] L. Shao, L. Liu, and M. Yu, "Kernelized multiview projection for robust action recognition," *International Journal of Computer Vision*, vol. 118, 06 2016.

[22] M. Baccouche, F. Mamalet, C. Wolf, C. Garcia, and A. Baskurt, "Sequential deep learning for human action recognition," 11 2011.

[23] J. Basavaiah and C. Patil, "Human activity detection and action recognition in videos using convolutional neural networks," *Journal of Information and Communication Technology*, vol. 19, pp. 157–183, 04 2020.

[24] B. Korbar, D. Tran, and L. Torresani, "Scsampler: Sampling salient clips from video for efficient action recognition," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 6232–6242, 2019.

[25] Z. Wu, C. Xiong, C.-Y. Ma, R. Socher, and L. S. Davis, "Adaframe: Adaptive frame selection for fast video recognition," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 1278–1287, 2019.

[26] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, "Image quality assessment: from error visibility to structural similarity," *IEEE Transactions on Image Processing*, vol. 13, pp. 600–612, 2004.

[27] A. Rahnama, A. Esfahani, and A. Mansouri, "Adaptive frame selection in two dimensional convolutional neural network action recognition," in *2022 8th Iranian Conference on Signal Processing and Intelligent Systems (ICSPIS)*, pp. 1–4, 2022.

[28] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural computation*, vol. 9, pp. 1735–80, 12 1997.

[29] K. Cho, B. van Merrienboer, Çaglar Gülçehre, D. Bahdanau, F. Bougares, H. Schwenk, and Y. Bengio, "Learning phrase representations using rnn encoder–decoder for statistical machine translation," in *Conference on Empirical Methods in Natural Language Processing*, 2014.

[30] C. Schuldt, I. Laptev, and B. Caputo, "Recognizing human actions: a local svm approach," in *Proceedings of the 17th International Conference on Pattern Recognition, 2004. ICPR 2004.*, vol. 3, pp. 32–36 Vol.3, 2004.

[31] Y. Shi, W. Zeng, T. Huang, and Y. Wang, "Learning deep trajectory descriptor for action recognition in videos using deep neural networks," pp. 1–6, 06 2015.

[32] V. Veeriah, N. Zhuang, and G.-J. Qi, "Differential recurrent neural networks for action recognition," pp. 4041–4049, 12 2015.

[33] H. Wang, A. Kläser, C. Schmid, and C.-L. Liu, "Action recognition by dense trajectories," *IEEE Conference on Computer Vision Pattern Recognition*, 06 2011.

[34] J. Liu, J. Luo, and M. Shah, "Recognizing realistic actions from videos "in the wild"," in *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1996–2003, 2009.

[35] W. Ahmed, M. H. Yousaf, and A. Yasin, "Robust suspicious action recognition approach using pose descriptor," *Mathematical Problems in Engineering*, 2021.

[36] J. Cho, M. Lee, H. J. Chang, and S. Oh, "Robust action recognition using local motion and group sparsity," *Pattern Recognit.*, vol. 47, pp. 1813–1825, 2014.

[37] G. K. Yadav, P. Shukla, and A. Sethfi, "Action recognition using interest points capturing differential motion information," in *2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 1881–1885, 2016.

[38] T. Shanableh, "Vico-moco-dl: Video coding and motion compensation solutions for human activity recognition using deep learning," *IEEE Access*, vol. 11, pp. 73971–73981, 2023.