Frame Differencing Based Temporal Feature Extraction in Human Action Recognition

S. Sowmyayani[1], V. Vivek[2]*, Arulpandy P[3], Tamilselvi S[4]

[1]Assistant Professor,Department of Computer Science (SF), St. Mary's College (Autonomous),Thoothukudi, Tamilnadu, India.
[2]Associate Professor, Department of Computer Science and Engineering, AAA College of Engineering and Technology,Sivakasi, Tamilnadu, India, Email: vivekmsucse@gmail.com
[3]Department of Mathematics, Christ University , Bangalore, Email: arulpandy.p@christuniversity.in
[4]Professor, Department of Biotechnology, Bannari Amman Institute of Technology India.
*Corresponding Author

Received: 05.04.2024 Revised : 08.05.2024 Accepted: 10.05.2024

ABSTRACT
Human Action Recognition (HAR) is applicable in many research domains such as video retrieval, autism care etc. Action can be recognized from the video whose content are temporal in nature. The two major challenges that HAR system faces are the temporal feature extraction and its computation cost. In this paper, these two challenges are rectified to some extent by introducing temporal feature extraction in residual frames. The Frame Differencing (FD) method is used to extract spatial and temporal features in order to recognize action. Keyframes are utilized to extract spatial features, whereas residual frames are utilized to recover temporal features. Both the features are fused to form spatio-temporal features and classified using Multiclass Support Vector Machine (MSVM) classifier. The proposed method is tested on HMDB51, UCF101 and UCF Sports datasets and the performance is measured using precision, recall, specificity and accuracy. It is also compared with most recent methods and found that the proposed method outperforms all compared methods by achieving an accuracy of 85.8%, 98.83% and 96.6% on HMDB51, UCF101 and UCF Sports action datasets respectively.

Keywords: spatial features, temporal features, keyframes, frame differencing

1. INTRODUCTION

From images to videos, human actions can be identified. In a content-based image and video retrieval system, it is one of the primary retrievals. The actions fall into two categories: slow and quick. Several action recognition datasets are available, including categories such as daily activities, sporting activities and more. From the video, spatio-temporal features are used to extract the action features. Over the past few decades, deep features and hand-crafted features have been used.

To extract features and categorize activities, a number of pre-trained models are also employed. One such model is Belief Network (BN). In order to enhance the BN, input data of various sizes, spatio-temporal Deep BN (DBN) and different pooling strategies are examined [1]. A sports action recognition method has also been developed based on particular spatio-temporal features. In [2], supervised action recognition model has been introduced for dark region. This work has used ARID (Action Recognition In the Dark) dataset for testing which recognizes actions even in the dark area.

A Convolutional Neural Network (CNN) based method has been developed for identifying human action in video and image [3]. This model is weak despite the robustness of the system as a whole. Another CNN-based technique with two layers for HAR has been described [4]. CNNs are trained to deliver information through video to an event that recognizes the value of the video at the first level. At the second level, they employ a Gated Recurrent Unit (GRU) and Long Short Term Memory (LSTM) based method to extract temporal and spatial data.

For action categorization, the SVM classification-based feature extraction method was created in [5]. But the accuracy of this approach is low. In order to model the video's subregions, spatio-temporal data are combined with Bag of Words (BoW) [6]. On the KTH, UCF sports, YouTube and Hollywood2 datasets, four dense trajectory features have been evaluated using the standard Bag of Features (BoF) [7].

From temporal frames in [8], the mixture of global and local Zernike moment features is extracted. A method for HAR based on a Deep Neural Network (DNN) and manually created features has been provided in [9]. This approach suffers from high computational time and complexity.

The suggested approach splits the video into different scenes. Keyframes are used to extract the spatial features of each scene, while intermediate frames are used to retrieve the temporal features. The Frame Differencing (FD) method is used to build residual frames from the intermediate frames. For the extraction of both spatial and temporal features, VGG16 is employed. For every scene, both features are merged and MSVM classifier is used for classification.

The main contributions of this paper include:

- The video is divided into scenes of different sizes. This gives temporal information for that particular scene alone.
- Spatial features are taken from the first frame of the scene. The residual frames are used to extract temporal features.
- The extraction of spatial features is done using pre-trained network model. Temporal feature extraction is performed using multiple pre-trained network models.

The rest of the document is arranged as follows: A few related papers that are used to examine more contemporary HAR research are covered in Section 2. The architecture and methods utilized in the suggested method are explained in detail in Section 3. Results and analysis of the experiment are shown in Section 4. Conclusion and future scope are provided in Section 5.

2. Related Works

Some of the most modern HAR techniques are covered in this section. Action recognition has been implemented for audio-enabled video data. Using unsupervised clustering in audio as a supervisory signal for video, a self-supervised Cross-Modal Deep Clustering (XDC) approach [10] has been created. The semantic correlation and the distinctions between audio and video have been utilized in this manner.

The extension of contrastive learning to a larger class of transformations and their combinations are studied in [11], wherein either uniqueness or invariance is necessary. The term Generalized Data Transformation (GDT) describes this technique. Some analyses are performed in [12] to enhance spatio-temporal 3D CNNs. It is investigated if the accuracy of video classification using spatio-temporal 3D CNNs would be enhanced by large-scale video datasets. According to the research, a thoroughly annotated dataset, such as Kinetics-700, may successfully pre-train a video representation for a video classification job.

In [13], dark videos are used to identify actions. In order to fill the data gap, this approach generates a fresh dataset for testing. In [14], a synthetic dataset of videos are created using 3D rendering tools and it is demonstrated that a classifier developed using this dataset could generalize to real videos. 3D convolution is combined with late temporal modeling in order to recognize activities [15].

A YouTube dataset for many viewpoints of outdoor activities has been described in [16]. Many research fields, such as action recognition, surveillance, etc., can benefit from this dataset. To improve the learning capacity of current self-supervised techniques, a Meta-Contrastive Network (MCN) [17] that combines contrastive learning with meta-learning has been designed.

Poisson distribution together with Univariate Measures (PDaUM), a feature selection technique, is created in [18]. In this strategy, a small percentage of the combined CNN features are redundant, leading to inaccurate predictions of intricate human movements. Because of this, this technique only chooses the most powerful features to provide to the Extreme Learning Machine (ELM) classifier. The keyframe segmentation approach is used in another study to distinguish activities [19]. In that technique, the entire video is utilised to extract temporal features using a Long Short Term Memory (LSTM) network, while features from keyframes are extracted using a MultiFiber Network (MFNet).

A different method that makes use of a hybrid SVM and K-Nearest Neighbor (KNN) classifier is predicated on a pre-trained deep CNN model [20]. It has been observed that previously learned CNN-based representations from a large annotated dataset can be transferred to a short training set action classification.

Unsupervised online deep learning approach has been used to construct a HAR method [21]. Action Recognition has been accomplished in [22] utilising local consistent group sparse coding with spatiotemporal structure. The depiction of convolutional maps on subsequent video frames has been processed using a convolutional GRU-Recurrent Neural Network (GRU-RNN) [23]. Utilizing discriminative structured trajectory groups, another approach has been developed [24].

In order to control the spatial and temporal kernels in different layers, factorized spatio-temporal CNNs are developed, which may lead to a decrease in the total number of learning parameters for the network [25]. Prior CNN-based methods were outperformed by the combined effect of the transformation and permutation operator, training and inference strategy, and a sparsity concentration index scheme. Certain researchers [26] claim that their Trajectory-pooled Deep-convolutional Descriptor (TDD) outperformed manually constructed features with a greater discriminating ability.

In this research, temporal feature extraction is done simply with residual frames. It would be superfluous to use the same keyframes that are solely used for spatial feature extraction. For temporal features, only the motion data is required.

3. Proposed FD-HAR Methodology

The proposed FD-HAR method consists of spatial feature extraction, temporal feature extraction and classification. This paper gives significant focus on temporal feature extraction. The proposed system architecture of FD-HAR method is shown in Fig. 1.

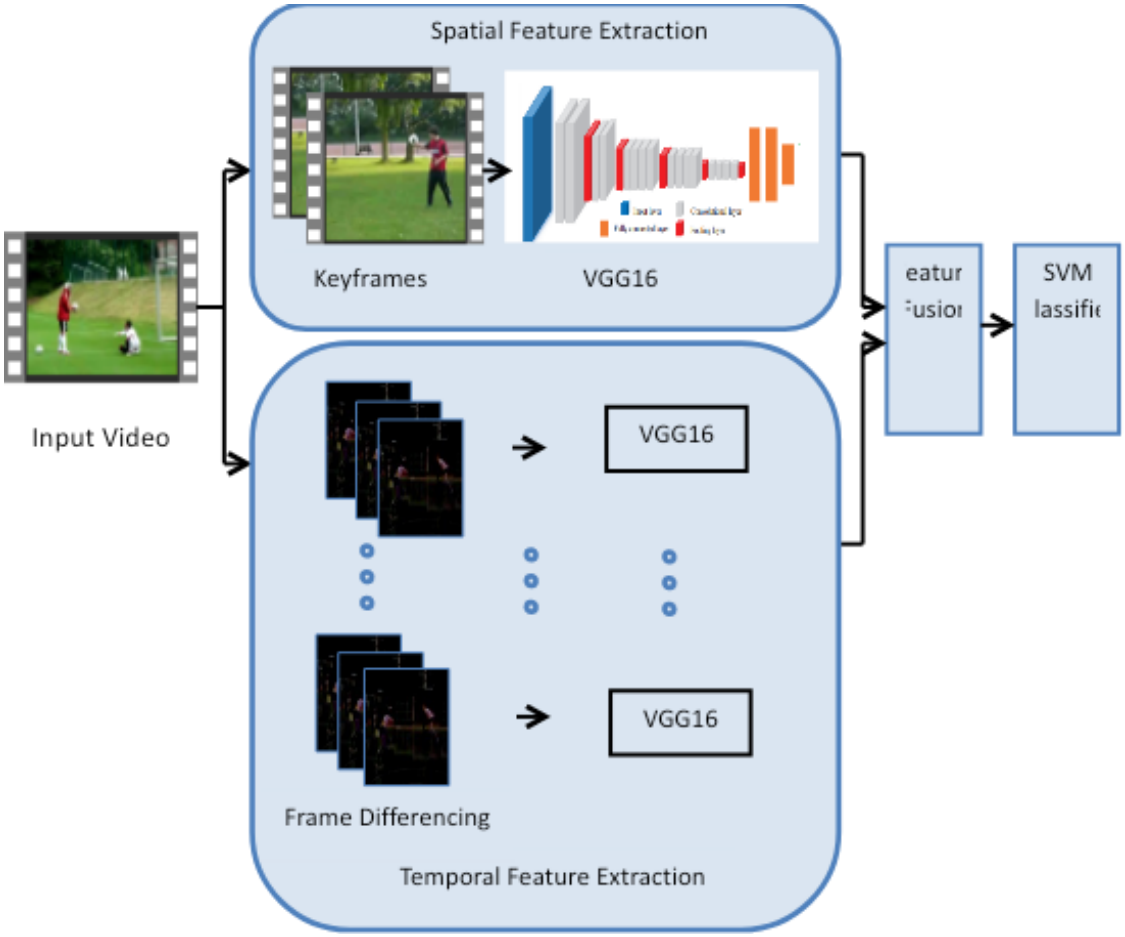

Fig 1. Proposed FD-HAR System Architecture

The video in the dataset is first separated into scenes, or Group of Pictures (GOP). Keyframes and intermediate frames make up each GOP. The frames that best reflect each GOP are called keyframes. The frames in between two consecutive keyframes are known as intermediate frames. Spatial feature extraction is applied to the keyframes. The FD approach is used to build residual frames from the intermediate frames [27]. The residual frames are used to extract temporal features. MSVM classifier is used after concatenating both features.

The spatial features are extracted based on scene content. Temporal features are extracted based on motion representation. It consists of Frame Differencing and Pooling. We use VGG-Net for extracting both features.

Scene Change based Segmentation (SCS) is solely used to extract keyframes from the video [28]. This method divides the incoming video into GOP segments. A single frame is chosen as the keyframe from each GOP. The first and last frames in every video are automatically designated as keyframes. It uses Pearson Correlation Coefficient (PCC) to identify the GOP. The PCC between frames f_k and f of size M x N is given as

$$PCC = \frac{\sum_{i=1}^{M} \sum_{j=1}^{N} (f_k(i,j) - f_k^m)((f(i,j) - f^m)}{\sqrt{\sum_{i=1}^{M} \sum_{j=1}^{N} (f_k(i,j) - f_k^m)^2 (f(i,j) - f^m)^2}} \quad (1)$$

Where f_k^m and f^m are the mean values of the two frames. From keyframes obtained from SCS, spatial features are retrieved using VGG16 network. The VGG-16 utilized in this work has 19 weight layers, made up of three Fully Connected (FC) layers and five blocks of convolutional layers (Conv.), with one pooling layer after each block [29]. The final FC layer's size is adjusted based on the number of classes and the input image size is shrunk to 224 × 224 × 3 in this design.

The FD approach is used in temporal feature extraction to get the motion information between keyframes K_f and intermediate frames I_f.

$$FD = |K_f - I_f| \quad (2)$$

Every GOP has two keyframes (the first and last frame in each GOP). The residual frame is the difference between the initial keyframe and the intermediate frame. The VGG16 network receives the set of residual frames in order to produce temporal features. The quantity of residual frames in each scene determines the size of the temporal feature for that scene. Similar to this, each scene's keyframe count—which is, of course, one—determines the size of each spatial feature. Therefore, to make the temporal features comparable with the spatial data, an average of the temporal information is obtained for each scene.

Algorithm 1 illustrates the integration of the techniques for extracting temporal and spatial data into a single procedure. In order to extract spatiotemporal features, the dataset is first split into training and testing sets. The procedures for extracting spatiotemporal features are as follows.

The dataset's videos are split up into frames. Keyframe status is set to the first frame. The following frame sequence indicates the keyframe that comes next. The PCC between the keyframe and the subsequent frames is computed for this purpose. The next keyframe was determined if the value of PCC exceeded a threshold. After it has been located, the difference between the keyframe and the intermediate frames are used to compute the residual frames.

.Algorithm: 1 Spatio-Temporal Feature Extraction

Input: Video V, VGG Network Model
Output: Spatio-temporal features
Steps:

1. Set $K_f = \{f_1\}$
2. $f_k = f_1$
3. **For** each frame f_i in V
4. Calculate PCC between f_k and f_i
5. **If** PCC >keyframe_threshold
6. Concatenate f_i into K_f
7. **For** each frame f_j between f_k and f_i
8. Calculate FD between f_j and f_i
9. Concatenate FD to FD_f
10. **End**
11. Give FD_f to VGG16 to get the features from the last FC layer
12. Find the average of the obtained features and concatenate to $f_{temporal}$
13. $f_k = f$
14. **Else**
15. Go to next frame
16. **End**
17. **End**
18. Give K_f to VGG16 to obtain the features from the last FC layer $f_{spatial}$
19. Concatenate $f_{spatial}$ and $f_{temporal}$ to create $f_{spatio\_temporal}$

After that, the pre-trained VGG16 network receives the residual frames in order to extract temporal features. Here, the VGG16 architecture that was employed for spatial feature extraction is employed. The final FC layer's features are extracted, and their average is determined. The temporal feature retrieved for a single scene is the acquired feature.

The last frame in the sequence is used to determine the final residual frames after all the keyframes have been located. In the end, the spatial features are recovered by feeding the VGG16 network with all of the keyframes as input. Let us denote the temporal features as $f_{temporal}$ and the spatial information retrieved from keyframes as $f_{spatial}$. The output features recovered from both VGGNets are of comparable sizes because each video was given the same amount of frames as for both spatial and temporal analysis. Therefore, combining the two properties column-wise is simple.

The keyframes in the method above are fed into VGG16 to extract spatial features. Similar to this, residual frames (FD_f) are the input for VGG16's temporal feature extraction. In Section 4, it is detailed how to analyze different PCC values to set the keyframe_threshold to 0.8. When a sequence has ten keyframes, the size of k_f also equals ten. Naturally, FD_f will have a size of 9. In order to match the size of the spatial characteristics, the final residual frame is concatenated once again.

The spatial and temporal features thus obtained are pooled to get a single feature for each sequence. The spatial and temporal features are concatenated column-wise as

$$f_{spatio-temporal} = [f_{spatial}\ f_{temporal}] \quad (2)$$

The size of $f_{spatio-temporal}$ is r x c where r is the number of sequences in the dataset and c is 8192 (which is size of feature extracted from fully connected layer of VGG16). Finally for each video sequences, the obtained spatial and temporal representations are then concatenated to train a non-linear MSVM classifier with a multichannel $\chi 2$ kernel.

4. Experimental Results

This section discusses the datasets used for testing the proposed method and the performance measures used to evaluate it. Then the hyperparameters used for training VGG16 and the results obtained by the proposed method are also discussed. The results are analyzed by comparing the proposed method with recent methods which is followed by ablation study.

Datasets and Performance Measures

The proposed method is tested on HMDB51 [30], UCF101 [31] and UCF Sports action [32] datasets. The HMDB51 dataset is very challenging than UCF101 dataset. HMDB51 consists of 6776 videos of 51 classes. In UCF101 dataset, there are 13320 videos of 101 classes. The videos in both the datasets have same resolution of size 320 x 240. The UCF sports action dataset (UCF Sports Website) contains 150 sequences of sport motions. Figure 2 shows sample frames from UCF sports action dataset.

Fig 2. Sample Frames from UCF Sports Action Dataset

The performance of the proposed HAR method is evaluated using specificity, precision, recall and accuracy which are given in Table 1.

Table 1. Performance Measures

Metrics	Formula
Precision	$\frac{TP}{TP + FP}$
Recall	$\frac{TP}{TP + FN}$
Specificity	$\frac{TN}{(FP + TN)}$
Accuracy	$\frac{TP + TN}{TP + TN + FP + FN} \times 100$

TP - True Positive, TN - True Negative, FP - False Positive, FN - False Negative

Hyperparameters and Results Analysis

The proposed action recognition method is executed in Nvidia Titan X GPU. The datasets are split into 80% training and 20% testing. The VGG16 network model is trained and tested with mini-batch size set

to 16 and dropout is set to 0.8. Adam Optimizer is used with learning rate is set to 10^{-3}. The performance of the proposed method is evaluated and the results are shown in Table 2.

Table 2. Results Obtained by the Proposed Method for all Compared Datasets

Dataset/Measure	Precision (%)	Recall (%)	Specificity (%)	Accuracy (%)
UCF101	98.35	97.54	98.8	98.8
HMDB51	85.1	84.87	85.9	86.8
UCF Sports	95.6	95.68	96	96.6

From Table 3, it is observed that the proposed method achieves above 95% precision, recall, specificity and accuracy on UCF101 and UCF sports action datasets. When datasets are compared, values are less on HMDB51 dataset. This is due to the fact that HMDB51 dataset is very challenging when compared to other datasets.

Comparison of Proposed Method with Recent Methods

The efficacy of the proposed method can be proved only when it is compared with other methods. Table 3 shows the comparison of proposed method with recent methods. The proposed action recognition method is compared with recent methods [10-18] that are discussed in Section 2.

Table 3. Accuracy Comparison of Proposed Method with Recent Methods on HMDB51 and UCF101 Datasets

Dataset/Method	HMDB51	UCF101
XDC [10]	65.1	94.2
GDT [11]	72.8	95.2
3D CNN [12]	69.4	92.9
Yuecong Xu et al. [13]	63.8	-
Matthews et al. [14]	83	-
Kalfaoglu et al. [15]	85.1	98.69
Perera et al. [16]	72.7	-
MCN [17]	54.8	85.4
PDaUM [18]	81.4	-
Proposed Methodology	86.8	98.8

From Table 3, it is observed that the proposed method achieves 86.8% and 98.8% accuracy on HMDB51 and UCF101 datasets respectively. In HMDB51 dataset, the proposed method achieves 1.7% higher accuracy than Kalfaoglu et al's method. Similarly, in UCF101 dataset, it achieves 0.11% higher accuracy than Kalfaoglu et al.'s method.

Table 4 compares the results of Allah BuxSargano et al.'s method [20], Charalampous and Gasteratos' method [21], Tian et al.'s method [22], Ballas et al.'s method [23], Atmosukarto et al.'s method [24], Sun et al.'s method [25], Wang et al.'s method [26] and the proposed method.

Table 4. Accuracy Comparison of Proposed Method with Recent Methods on UCF Sports Action Dataset

Author, Year	Accuracy (%)
Allah BuxSargano et al., [20]	91.47
Charalampous and Gasteratos, [21]	88.55
Tian et al., 2016 [22]	90.0
Ballas et al., [23]	80.7
Atmosukarto et al., [24]	82.6
Sun et al., [25]	88.1
Wang et al., [26]	95.1
Proposed Method	96.6

From Table 4, in UCF sports action dataset, Wang et al.'s method [26] outperforms other methods by more than 3%. The proposed method obtains 96.6% accuracy which is greater than Wang et al.'s method [26].

Ablation Study

The proposed method is analyzed for various PCC threshold values. The value of PCC lies between 0 (no correlation) and 1 (highly correlated). For splitting the video into scenes, PCC is used in this work. The value of PCC is varied from 0.3 to 0.8. Remaining values are left out, as it is not useful for our work.

A sample video is taken for this analysis whose scenes are identified manually. The number of frames for each scenes obtained by the proposed method is compared with manual scene identification. The number of frames in each scene identified manually and PCC based scene identification is given in Fig. 3.

Manual Scene Identification	
Scenes	**No. of frames**
Scene 1	10
Scene 2	12
Scene 3	9
Scene 4	15

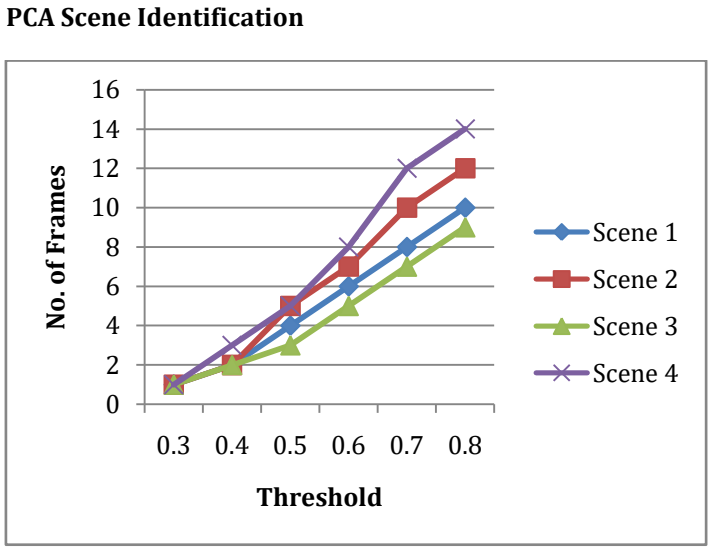

Fig 3. Comparison of Manual and PCA Based Scene Identification

From Fig. 3, it is clear that the number of frames in every scene is correctly identified when keyframe threshold is set to 0.8. Hence this threshold is set for keyframe selection.

The proposed method is also tested with AlexNet instead of VGG16. The other networks suffer from heavy computation time. Hence, only AlexNet is used for analysis. Table 5 shows the accuracy achieved by the proposed method for both networks with the same hyperparameters.

Table 5. Accuracy Obtained by the Proposed Method for different Network Models

Network Model	HMDB 51	UCF101	UCF Sports
AlexNet	84.7	96.8	95.3
VGG16	85.6	98.71	96.6

From Table 5, it is evident that the proposed method with VGG16 network outperforms AlexNet with a little increase in accuracy.

CONCLUSION

The human action recognition dataset's films are categorized into scenes using varying numbers of frames in the suggested technique. Keyframes are utilized to extract the spatial features in each scene, whereas intermediate frames are used to retrieve the temporal features. Keyframes are used to transform the intermediate frames into residual frames. MultiSVM classifier is used to concatenate and classify both the spatial and temporal information. UCF101 and HMDB51, two publicly accessible datasets, are used to test this approach. On the HMDB51 and UCF101 datasets, it obtains accuracy of 86.8% and 98.8%, respectively, which is greater than recent techniques.

REFERENCES

[1] Guo, Y. and Wang, X., 2021. Applying TS-DBN model into sports behavior recognition with deep learning approach. The Journal of Supercomputing, pp.1-17.

[2] Hira, S., Das, R., Modi, A. and Pakhomov, D., 2021. Delta Sampling R-BERT for Limited Data and Low-Light Action Recognition. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (pp. 853-862).

[3] Hara, K.; Kataoka, H.; Satoh, Y. Can spatiotemporal 3d cnns retrace the history of 2d cnns and imagenet? In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018; pp. 6546–6555
[4] Zhang, L.; Xiang, X. Video event classification based on two-stage neural network. Multimed. Tools. Appl. 2020, 1–16.
[5] Meng, Q.; Zhu, H.; Zhang, W.; Piao, X.; Zhang, A. Action Recognition Using Form and Motion Modalities. ACM Trans. MCCA 2020
[6] Shapovalova, N., Vahdat, A., Cannons, K., Lan, T., Mori, G.: Similarity constrained latent support vector machine: an application to weakly supervised action classification. In: European Conference on Computer Vision, pp. 55–68. Springer (2012)
[7] Wang, H., Kläser, A., Schmid, C., Liu, C.L.: Dense trajectories and motion boundary descriptors for action recognition. Int. J. Comput. Vis. 103(1), 60–79 (2013)
[8] Aly, S., Sayed, A.: Human action recognition using bag of global and local zernike moment features. Multimed. Tools Appl. 1–31 (2019)
[9] Khan Sharif M, Akram T, Raza M, Saba T, Rehman A (2020) Hand-crafted and deep convolutional neural network features fusion and selection strategy: an application to intelligent human action recognition. Appl Soft Comput 87:105986
[10] HumamAlwassel, Bruno Korbar, Dhruv Mahajan, Lorenzo Torresani, Bernard Ghanem, and Du Tran. Self-supervised learning by cross-modal audio-video clustering. In NeurIPS, 2020
[11] Patrick, M., Asano, Y.M., Kuznetsova, P., Fong, R., Henriques, J.F., Zweig, G. and Vedaldi, A., 2021. On compositions of transformations in contrastive self-supervised learning. In Proceedings of the IEEE/CVF International Conference on Computer Vision (pp. 9577-9587).
[12] Kataoka, H., Wakamiya, T., Hara, K. and Satoh, Y., 2020. Would mega-scale datasets further enhance spatiotemporal 3D CNNs?. arXiv preprint arXiv:2004.04968.
[13] YuecongXu ,Jianfei Yang , Haozhi Cao , Kezhi Mao , Jianxiong Yin and Simon See, 2020 "ARID: A Comprehensive Study on Recognizing Actions in the Dark and A New Benchmark Dataset", arXiv preprint arXiv: 2006.03876
[14] Matthews, O., Ryu, K. and Srivastava, T., 2020. Creating a Large-scale Synthetic Dataset for Human Activity Recognition. arXiv preprint arXiv:2007.11118.
[15] Kalfaoglu, M.E., Kalkan, S. and Alatan, A.A., 2020, August. Late temporal modeling in 3d cnn architectures with bert for action recognition. In European Conference on Computer Vision (pp. 731-747). Springer, Cham.
[16] Perera, A.G., Law, Y.W., Ogunwa, T.T. and Chahl, J., 2020. A multiviewpoint outdoor dataset for human action recognition. IEEE Transactions on Human-Machine Systems, 50(5), pp.405-413.
[17] Lin, Y., Guo, X. and Lu, Y., 2021. Self-supervised video representation learning with meta-contrastive network. In Proceedings of the IEEE/CVF International Conference on Computer Vision (pp. 8239-8249).
[18] Khan, M.A., Zhang, Y.D., Khan, S.A., Attique, M., Rehman, A. and Seo, S., 2020a. A resource conscious human action recognition framework using 26-layered deep convolutional neural network. Multimedia Tools and Applications, pp.1-23.
[19] Chen, Y., Kalantidis, Y., Li, J., Yan, S. and Feng, J., 2018. Multi-fiber networks for video recognition. In Proceedings of the european conference on computer vision (ECCV) (pp. 352-367).
[20] Allah BuxSargano, Xiaofeng Wang, Plamen Angelov, and Zulfiqar Habib, "Human Action Recognition using Transfer Learning with Deep Representations, IEEE, 2017, pp. 463-469.
[21] Charalampous, K. and A. Gasteratos, On-line deep learning method for action recognition. Pattern Analysis and Applications, 2016. 19(2): p. 337-354.
[22] Tian, Y., Ruan, Q., An, G., Fu, Y., Action Recognition Using Local Consistent Group Sparse Coding with Spatio-Temporal Structure. in Proceedings of the 2016 ACM on Multimedia Conference. 2016. ACM.
[23] Ballas, N., L. Yao, C. Pal, and A. Courville, "Delving deeper into convolutional networks for learning video representations," International Conference of Learning Representations, 2016.
[24] Atmosukarto, I., N. Ahuja, and B. Ghanem. Action recognition using discriminative structured trajectory groups. in 2015 IEEE Winter Conference on Applications of Computer Vision. 2015. IEEE
[25] Sun, L., K. Jia, D.-Y. Yeung, and B. E. Shi, "Human action recognition using factorized spatio-temporal convolutional networks," in Proceedings of the IEEE International Conference on Computer Vision, 2015, pp. 4597–4605.
[26] Wang, L., Y. Xiong, Z. Wang, and Y. Qiao, "Towards good practices for very deep two-stream convnets," in arXiv:1507.02159, 2015.
[27] Sowmyayani, S. and Rani, P.A.J., 2016. Frame differencing-based segmentation for low bit rate video codec using H. 264. International Journal of Computational Vision and Robotics, 6(1-2), pp.41-53.

[28] Sowmyayani, S., Rani, P. A. J., 2014. Adaptive GOP structure to H. 264/AVC based on Scene change. ICTACT Journal on Image & Video Processing, 5(1).
[29] Simonyan K, Zisserman A. Very deep convolutional networks for large scale image recognition. In: Proceedings of International Conference on Learning Representations (ICLR), San Diego, 2015. 1–14
[30] K. Soomro, A. R. Zamir, and M. Shah. UCF101: A dataset of 101 human actions calsses from videos in the wild. Technical Report CRCV-TR-12-01, 2012.
[31] H. Kuehne, H. Jhuang, E. Garrote, T. Poggio, and T. Serre. HMDB: a large video database for human motion recognition. In Proc. ICCV, 2011.
[32] UCF Sports Website: http://crcv.ucf.edu/data/UCF Sports Action.php