TL-SMMSS: Transfer Learning with Stacked Mean of Max SVM SoftMax Layer for Content-Based Action Video Retrieval

Alina Banerjee[2*], Ravinder M[1†]

[1*]Department of Computer Science and Engineering, Indira Gandhi Delhi Technical University for Women, New Church Rd, New Delhi, 110006, India.
[2]School of Engineering and Technology, K R Mangalam University University, Sohna, 122103, Haryana, India.

*Corresponding author(s). E-mail(s): alina.chak@gmail.com; Contributing authors: ravinderm@igdtuw.ac.in; †These authors contributed equally to this work.

Abstract

In multimedia analysis, action video retrieval is a significant challenge that requires effective techniques to locate and extract relevant video information from large datasets accurately. This research introduces a novel method for action video retrieval known as Transfer Learning with Stacked Mean of Max SVM SoftMax Layer (TL-SMMSS). The proposed approach enables effective feature extraction from video frames by leveraging transfer learning with pre-trained deep learning models. It combines the strengths of SoftMax layers and Support Vector Machines (SVM) in a unique layered architecture. Specifically, frame-level features are aggregated into a compact video-level representation. The SoftMax layer ensures probabilistic output for retrieval ranking, while the SVM layer is incorporated to improve classification robustness. Experimental results on benchmark action video datasets demonstrate that TL-SMMSS outperforms state-of-the-art techniques in both computational efficiency and retrieval accuracy. The method presents a mean average precision of 0.849 and 0.6105 for the UCF101 and HMDB51 datasets, respectively. This method provides a scalable and effective solution for retrieving action videos, with potential applications in multimedia search engines, sports analysis, and video surveillance.

Keywords: Transfer Learning; Mean of Max SVM; SoftMax; Video Retrieval

Introduction

Early systems that used manually created features, such as color histograms and motion vectors, failed to connect low-level data with high-level semantics. This led to the development of content-based video retrieval (CBVR) [1]. Although it required a lot of processing power and big labeled datasets, machine learning enhanced feature representation. Although training deep models from scratch still requires a lot of resources, the emergence of deep learning, particularly CNNs, significantly improved CBVR by capturing intricate spatial-temporal patterns [2]. The large dimensionality and heterogeneity of video data, which includes differences in quality, frame rate, and modality, provide significant hurdles for CBVR [3]. It

is difficult to extract significant features while maintaining real-time performance, which emphasizes the necessity of effective techniques like transfer learning to overcome data and computational constraints.

Transfer learning has emerged as a powerful paradigm for mitigating the challenges of data scarcity and high computational demands in CBVR. The central idea of transfer learning is to leverage knowledge gained from solving one problem and apply it to a related, yet different, task [4]. In the context of CBVR, transfer learning often involves pre-trained deep learning models trained on large datasets, such as ImageNet for images or Kinetics for videos. These models capture general features, such as edges and textures, in their initial layers, while their deeper layers encode more task-specific representations. By fine-tuning these pre-trained models on domain-specific video datasets, researchers can achieve high performance with significantly less labeled data and computational cost. Transfer learning not only accelerates the development of CBVR systems but also enables them to generalize better across diverse video datasets.

Pre-trained models are the backbone of transfer learning in CBVR tasks. ResNet, VGG, and Inception, trained on the ImageNet dataset, have become standard for feature extraction in video retrieval systems [5]. For video-specific tasks, models trained on datasets like Kinetics or Sports-1M provide temporal and motion-aware representations, making them highly suitable for tasks like action recognition and event detection [5]. Recently, transformer-based architectures such as Vision Transformers (ViTs) and Video Swin Transformers have demonstrated remarkable performance in video understanding by capturing both spatial and temporal dependencies more effectively [6]. These pre-trained models offer a robust starting point for CBVR applications, allowing researchers to fine-tune them for specific use cases with minimal computational effort.

Transfer learning has enabled a wide range of applications in CBVR, addressing some of its most pressing challenges. One notable application is action recognition, where pre-trained models fine-tuned on datasets like UCF101 or HMDB51 achieve state-of-the-art performance in identifying human actions in videos [7]. Similarly, transfer learning facilitates Another significant application is event detection, where CBVR systems identify and retrieve video segments corresponding to specific events, such as goal scoring in sports or unusual activities in security footage. Beyond these, transfer learning has been instrumental in developing personalized video recommendations, where models adapt to user preferences by leveraging pre-trained feature representations [8]. These applications underscore the versatility and efficiency of transfer learning in advancing the capabilities of CBVR systems.

While transfer learning has significantly advanced the field of content-based video retrieval (CBVR), its implementation and optimization are not without challenges. Below are the key issues encountered in transfer learning and fine-tuning for video retrieval.

Domain Gap Between Source and Target Data. Transfer learning involves leveraging knowledge from a pre-trained model, which is often trained on a generic dataset like ImageNet or Kinetics. However, a domain gap can exist between the source dataset (pre-training) and the target video dataset. This discrepancy in visual, temporal, or contextual characteristics may reduce the effectiveness of feature transfer, leading to suboptimal performance [9]. For instance, models pre-trained on human-centric actions may struggle with nonhuman video datasets, such as wildlife monitoring or industrial processes.

Computational Complexity Video data is inherently high-dimensional, involving spatial, temporal, and sometimes audio components. Finetuning models on such data requires significant computational resources, including memory and processing power. This is especially challenging for large transformer-based architectures, which have a high computational cost [10]. The need to process long video sequences exacerbates this issue, making real-time video retrieval tasks computationally demanding.

Temporal Feature Learning Unlike static images, videos require the model to capture temporal dynamics across frames. While transfer learning models like ResNet or Vision Transformers are adept at extracting spatial features, they may not be optimized for temporal relationships in video data. Integrating temporal modeling into fine-tuning workflows remains a complex task [7].

Dataset-Specific Feature Representation Some video retrieval tasks require highly specialized features, such as motion cues for sports analysis or texture patterns for medical videos. Pre-trained models may lack the capability to represent such domain-specific features

effectively, necessitating additional architectural modifications or fine-tuning strategies [11].

Evaluation Complexity Assessing the performance of transfer learning and fine-tuning for video retrieval is non-trivial. Video retrieval tasks often involve subjective quality metrics, such as relevance or interpretability, in addition to standard quantitative metrics like accuracy or mean Average Precision (mAP). These subjective metrics are challenging to standardize, complicating model evaluation [12].

The contributions of the research are as follows:

- In this research for addressing the domain gap between source and target data 3D convolutional Resnets pre-trained on Kinetics 700 [13] have been utilized as the backbone network.
- For addressing computational complexity transfer learning has been utilized by freezing all the topmost layers, since finetuning of such models requires huge computational resources including memory and processing power.
- For addressing the data imbalance problem of video datasets and convergence time of classifiers a Hybrid Mean of Max SVM Softmax activation function has been proposed.
- The proposed transfer learning network with the hybrid softmax activation function has been tested on UCF101 and HMDB51 datasets for video retrieval tasks.

The related studies on this topic that are covered in the remainder of the paper are introduced in Section 2. In Section 3, the visual similarity function employing similarity measures is explained and a top-k query is built. The suggested method of Transfer Learning with hybrid Mean of Max SVM Softmax activation is explained in Section 4. Section 5 presents the data set that is used for the experimentation, results and outcomes. Section 6 presents the evaluation metrics. Section 7 concludes the paper.

Related Work

Content-based video retrieval (CBVR) has witnessed significant advancements with the advent of deep learning and transfer learning techniques. This review discusses the evolution and application of various transfer learning networks in CBVR, focusing on their architectures, datasets, and performance benchmarks. Adopting transfer learning in CBVR initially relied on CNNs pre-trained on image datasets like ImageNet. Networks such as AlexNet and VGG were repurposed for video tasks by treating individual frames as static images [14]. However, the lack of temporal modeling limited their applicability. The primary strategy was to freeze the early layers of these models, which encoded generic visual features, and fine-tune the deeper layers for specific video retrieval tasks. Two-stream networks introduced a significant advancement by explicitly addressing the temporal dimension in videos. The researchers [14][11] proposed a model combining a spatial stream (RGB frames) with a temporal stream (optical flow), which was trained separately and fused at the decision level. These networks utilized pre-trained weights from image datasets for the spatial stream, while the temporal stream was initialized with random weights or pre-trained on video datasets like UCF101. While effective for action recognition, their computational demands limited scalability in CBVR systems. To overcome the limitations of 2D CNNs, researchers developed 3D CNNs, such as C3D and I3D, which process spatiotemporal data directly [7][15]. I3D extended the idea of 2D CNNs by inflating them to operate on 3D convolutions, initializing weights from 2D ImageNet models. This approach significantly improved video understanding and retrieval accuracy by modeling spatial and temporal dependencies jointly. However, the computational overhead of 3D CNNs remained a concern, particularly for large-scale CBVR tasks. Transformer-based architectures have recently gained traction in CBVR due to their ability to model long-range dependencies. Vision Transformers (ViT), pre-trained on large-scale datasets, demonstrated superior performance in image-based tasks and were adapted for video retrieval through fine-tuning [16]. Video Swin Transformers [17] extended this concept by incorporating spatiotemporal attention mechanisms, achieving state-of-the-art results on video retrieval benchmarks. These models rely heavily on transfer learning due to their large parameter space and require substantial pre-training resources. Hybrid approaches combining CNNs and transformers have been explored to balance efficiency and

performance. For instance, R(2+1)D networks factorize 3D convolutions into 2D spatial and 1D temporal convolutions, initializing spatial layers with ImageNet weights and temporal layers with video datasets [18]. This strategy leverages transfer learning while reducing computational complexity. Similarly, models that integrate LSTMs with pre-trained CNNs capture sequential patterns in video frames, enhancing retrieval accuracy in CBVR systems [19]. Despite their success, transfer learning networks face challenges such as domain adaptation, overfitting during fine-tuning, and computational scalability. Recent studies focus on unsupervised pre-training and domain-specific adaptations to address these issues. Contrastive learning methods, such as those implemented in MoCo and SimCLR, have also been adapted for video tasks to pre-train models in an unsupervised manner, enabling more effective transfer learning for CBVR [20]. Table I depicts the Transfer Learning Convolutional Networks used in the video retrieval tasks with their challenges:

Problem Statement

Content-based video retrieval (CBVR) involves retrieving videos from a large database based on the visual content in a query video or a query frame. The goal is to automatically search and rank videos in a database that are visually similar or relevant to a given query video or frame. Traditional video retrieval methods often rely on handcrafted features such as colour histograms, motion features, or texture, but these methods are limited in performance and flexibility. Transfer learning, using pre-trained deep neural networks (especially convolutional neural networks), offers a powerful approach by enabling the extraction of rich, high-level features from videos that can be used for content-based retrieval tasks. In this context, we leverage pre-trained models for feature extraction and fine-tune them to adapt to the specific video retrieval task. The objective is to formulate a solution for content-based video retrieval using transfer learning. The goal is to identify and retrieve relevant videos from a large database based on the similarity to a query video, using a pre-trained deep learning model and adapting it to the specific domain of video content.

Proposed Framework

A technique for transfer learning with a pre-trained 3D convolutional network (R3D50) [13] for video classification is proposed, where we replace the classification head with a hybrid Meanof Max SVM SoftMax classifier instead of simple SoftMax- activation. We use a 3D convolutional model (R3D50) pre-trained on video data Kinetics-700[13]. The model is loaded and its final classification layers are removed and the global average pooling layer is replaced with the max pooling layer since the max pooled layers give higher mean average precision in retrieval in the datasets UCF101 and HMDB51. So, this modified structure of R3D50 is used as the feature extraction layer. Then the feature extraction layer is connected to the hybrid Mean of Max SVM SoftMax layer. The architecture described is a modified version of the 3D ResNet-50 model designed for tasks involving spatiotemporal data, such as video classification. Below is a detailed explanation of the model architecture and modifications. The 3D ResNet-50 employs 3D convolutions to extract spatiotemporal features from input data. Unlike 2D CNNs, 3D CNNs capture motion and spatial patterns simultaneously by considering the temporal dimension in addition to height and width. The feature extraction layers, which consist of the initial convolutional layers and ResNet-50's residual blocks, are frozen during training. Freezing these layers prevents them from being updated, retaining their pre-trained weights. This is useful when transferring knowledge from large datasets like Kinetics or Sports1M to new, smaller datasets. After the feature extraction, a 3D max-pooling layer is applied. Max pooling reduces the spatial and temporal dimensions, focusing on the most salient features. This pooling operation enhances robustness to spatial and temporal variations in the input data. The output of the

Table 1 Transfer Learning Networks in Content-Based Video Retrieval

Convolutional Backbone	Pre-trained Dataset	Features	Retrieval Dataset	Challenges
VGG [21][11]	ImageNet	RGB frames and optical flow	UCF101, HMDB51	High computational cost due to separate stream training
VGG [22]	ImageNet	Combined deep features with trajectory-based descriptors	UCF101, HMDB51	Dependency on handcrafted features
Alexnet [19]	ImageNet	Spatial features combined with LSTM	UCF101, Sports1M	Expensive Computational Resource Requirement
R3D18[23]	Kinetics 400	Deep spatiotemporal features with Bidirectional LSTM	UCF101, HMDB51, JHMDB	Expensive memory and computational requirement
C3D [24][25]	Sports1M	RGB frames	UCF101, HMDB51	Expensive computational requirement
R3D, R (2+1) D [26]	Kinetics 400	RGB frames	UCF101, HMDB51	Retrieval results are not as promising as many benchmark techniques

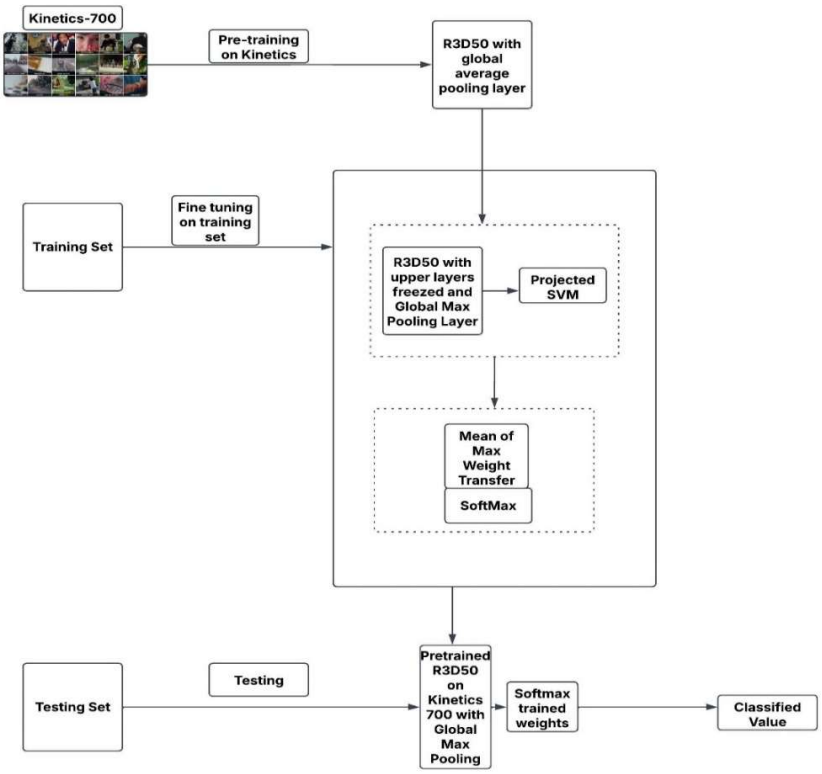

Figure 1 Workflow of the Transfer Learning Process with Mean of Max SVM SoftMax Layer

maximum pooling layer is flattened into a feature vector. The SVM layer inspired by authors [27] is added after the max-pooling layer. This layer separates the classes of multiclass SVMs by Euclidean projection onto the positive simplex [27]. A SoftMax layer follows the SVM, transferring the SVM's weights to it. This layer makes the model output interpretable by providing class probabilities, which is helpful for multiclass classification tasks. The iteration of the SVM is decided by checking the accuracy of the training set. Once the accuracy of the training set is set to the maximum value for a certain number of iterations, the corresponding weight of the SVM classifier is transferred to the SoftMax activation function and run until the requisite number of iterations is reached, until the accuracy of the training set is maximized.

4.1 Time Complexity Analysis of the Mean of Max SVM Softmax Regression

The Projected SVM [27] has the number of iterations as K = 100, then the mean of max accuracy calculation of the training set is having a constant value C. Accordingly, the average of max accuracy weights and biases are calculated in constant time and then the weights and biases are transferred to the weights of softmax regression with K1 = 400 iterations. So the number of iterations or epochs, since the classifiers run on the whole training set at once is K+K1+C.

5 Retrieval Metrics

Retrieval metrics in content-based video retrieval (CBVR) using transfer learning are essential for evaluating system performance and ensuring effective retrieval of relevant video content. Common metrics include mean Average Precision (mAP) [13], which calculates the average precision across all queries to assess relevance, and Precision at k, which measures the proportion of relevant videos in the top k retrieved results [30]. Other metrics such as Recall is also employed to evaluate the system's ability to rank relevant content effectively. Transfer learning enables these metrics to benefit from pre-trained models by leveraging rich feature representations from large-scale datasets, improving retrieval accuracy even with limited target-domain data [31]. These metrics provide a comprehensive view of retrieval effectiveness and are widely used in CBVR research. The mAP is computed using Equation 1.

$$mAP = \frac{\sum_{q=1}^{n} AP(q)}{Q} \quad (1)$$

Where

$$Q = Number\ of\ Query\ Videos \quad (2)$$

and the average precision function (AP) is defined in Equation 3.

$$AP = \frac{\sum_{k=1}^{n}(P(k) \times rel(k))}{Number\ of\ Relevant\ Video} \quad (3)$$

where P is the precision function that returns the cut-off k precision [13]
rel is the masking function that returns 1 if the video prototype at k is relevant or 0 otherwise [13].

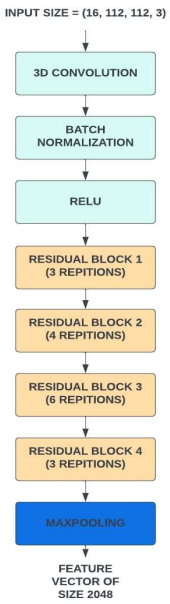

Figure 2 Frozen Layers of R3D50 used for Transfer Learning

Algorithm 1 Algorithm of Mean of Max SVM SoftMax Regression

```
f (X, SVM_iter, SoftMax_iter):
Xtrain = Feature_Extractor (X) as given Figure2
Wts = [] > List to store the weights where the training accuracy is maximized
dt = [] > List to store delta values where training accuracy is minimized
acc = [] > List to store training set accuracy for projected SVM [23]
while i <= SVM_iter do
   if accuracy_score(Xtrain) == 1
     Wts.add(projected_SVM_weights[23])
     dt.add(projected_SVM_delta[23])
     acc.add(Accuracy_trainingset[23])

W = Mean (Wts)
B = Mean(dt)

net_input (Xtrain, W, b):
return (dotproduct(Xtrain, W)+b)

msoftmax(z):
return e^(dotproduct(z.Transpose)) / sum(e^(z)).Transpose
softmax(z):
return (e^((z.Transpose−m (z)) / sum(e^((z−max (z)))) .Transpose

to_classlabel(z):
return argmax(z)        >maximum argument of z
global W, B             > W = Weight Matrix, B = Bias matrix
eta = 0.01
l2 = 0.001
W = W
B = B
N = Number of training data samples
Xtrain = Training Set
While i <= SoftMax_iter do
net = net_input(Xtrain, W, B)
softm = softmax(net)
diff = (softm – y_enc)
grad = (1/N) x dot (Xtrain.Transpose, diff)
W -= (eta x grad + 2 x eta x l2 x W)
B -= (eta x sum(diff))
net = net_input(Xtrain, W, B)
softm = softmax(net)
smax = to_classlabel(softm)
S = accuracy_score(data_train, smax)
i ++
end While
return S, W, B
```

Datasets, Results and Outcomes

The mAP of the proposed Mean of Max SVM-SoftMax Transfer Learning-based classified values is compared with benchmark weight initialization techniques, stacking and ensembling techniques, and Transfer Learning for video retrieval. The approach is implied to provide better top-1 accuracy when retrieving video database prototypes as depicted in Figure 1. The approach produces top 1 accuracy values of 0.6105 and 0.849 for the HMDB51[29] and UCF101[28] datasets, respectively. Accuracy among the top 5 has decreased. Therefore, better classifier performance would lead to increased retrieval accuracy for the top 1. The technique has been implemented with the following CPU specifications: Intel(R) Xeon(R) CPU @ 2.20GHz. The results generated are from the experimental values of SVM iter = 100, and SoftMax iter = 400 as defined in section 4. For the HMDB51 dataset and UCF101, the suggested classifier provides the performance specified at 500 epochs and learning rate = 0.01 and L2 = 0.001. Table 3 compares the benchmark transfer learning techniques using pre-trained convolutional networks.

Table 3 Comparison **of** Accuracy of Proposed Technique with Benchmark Transfer Learning Techniques using pre-trained convolutional networks utilized in video retrieval

Datasets	Learning Techniques	Pre-trained network	Pre-trained on	Top1 accuracy
UCF101	Supervised Video Hashing [23]	R3D18	Kinetics-400	0.806
	Feature selection and hash code generation [24]	C3D	Sports1M	0.8321
	Temporal Graph Learning [26]	C3D	Sports1M	0.2813

	Prototype Learning [32]	R3D50	Kinetics-700	0.84
	Hybrid SVM-Softmax [33]	R3D50	Kinetics-700	0.844
	Proposed Technique	R3D50	**Kinetics-700**	**0.849**
HMDB51	Supervised Video Hashing [23]	R3D18	Kinetics-400	0.575
	Feature selection and hash code generation [24]	C3D	Sports1M	0.75
	Temporal Graph Learning [26]	C3D	Sports1M	0.07
	Prototype Learning [32]	R3D50	Kinetics-700	0.562
	Hybrid SVM-Softmax [33]	R3D50	Kinetics-700	0.583
	Proposed Technique	R3D50	Kinetics-700	0.6105

Conclusion and Future Scope

Action video retrieval is a major problem in multimedia analysis that calls for efficient methods to precisely find and extract pertinent video information from big datasets. This study presents Transfer Learning with Stacked Mean of Max SVM SoftMax Layer (TL-SMMSS), a novel approach for action video retrieval. The suggested method uses transfer learning with pre-trained deep learning models to enable efficient feature extraction from video frames. It creates a special layered architecture by fusing the advantages of Support Vector Machines (SVM) with SoftMax layers. In particular, a condensed video-level representation is shaped by combining frame-level characteristics. While the SVM layer is included to increase classification resilience, the SoftMax layer guarantees probabilistic output. Results from experiments on benchmark action video datasets show that TLSMMS performs better than the most advanced methods in terms of retrieval accuracy and computing efficiency. For the UCF101 and HMDB51 datasets, the method's mean average precision is 0.849 and 0.6105, respectively. With potential uses in sports analysis, video surveillance, and multimedia search engines, this technique offers a scalable and efficient way to retrieve action recordings. Further studies can be conducted on the above method by modifying the feature extraction layers of the pre-trained network. Also, optimized structures of the classification layers can be proposed.

Declarations

- **Compliance with Ethical Standards** The article lacks any studies conducted by the authors involving human participants or animals.
- **Funding**
None
- **Conflict of interest** None
- **Availability of data and materials** The datasets analysed during the current study are available in the repository with Weblink [https://www.crcv.ucf.edu/data/UCF101/ UCF101.rar]. [http://serre-lab.clps.brown.edu/wpcontent/uploads/2013/10/hmdb51_org.rar]

References

[1] Smeulders, A.W.M., Worring, M., Santini, S., Gupta, A., Jain, R.: Content-based image retrieval at the end of the early years. IEEE Transactions on Pattern Analysis and Machine Intelligence **22**(12), 1349–1380 (2000) https://doi.org/10. 1109/34.895972

[2] Krizhevsky, A., Sutskever, I., Hinton, G.E.: Imagenet classification with deep convolutional neural networks. Communications of the ACM **60**(6), 84–90 (2017) https://doi.org/10.1145/3065386

[3] Hampapur, A., Brown, L., Connell, J., Ekin, A., Haas, N., Lu, M., Merkl, H., Pankanti, S.: Smart video surveillance: exploring the concept of multiscale spatiotemporal tracking. IEEE Signal Processing Magazine **22**(2), 38–51 (2005) https://doi.org/10.1109/msp.2005.1406476

[4] Pan, S.J., Yang, Q.: A survey on transfer learning. IEEE Transactions on Knowledge and Data

Engineering **22**(10), 1345–1359 (2010) https://doi.org/10.1109/ tkde.2009.191

[5] Kay, W., Carreira, J., Simonyan, K., Zhang, B., Hillier, C., Vijayanarasimhan, S., Viola, F., Green, T., Back, T., Natsev, P., Suleyman, M., Zisserman, A.: The Kinetics Human Action Video Dataset (2017). https://arxiv.org/abs/1705.06950

[6] Zhang, B., Ma, R., Cao, Y., An, P.: Swin-vec: Video swin transformer-based gan for video error concealment of vvc. The Visual Computer **40**(10), 7335–7347 (2024) https://doi.org/10.1007/s00371-024-03518-9

[7] Carreira, J., Zisserman, A.: Quo vadis, action recognition? a new model and the kinetics dataset. In: 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). IEEE, (2017). https://doi.org/10.1109/cvpr.2017.502 . http://dx.doi.org/10.1109/cvpr.2017.502

[8] Covington, P., Adams, J., Sargin, E.: Deep neural networks for youtube recommendations. In: Proceedings of the 10th ACM Conference on Recommender Systems. RecSys '16, pp. 191–198. ACM, (2016). https://doi.org/10.1145/2959100.2959190 . http://dx.doi.org/10.1145/2959100.2959190

[9] Weiss, K., Khoshgoftaar, T.M., Wang, D.: A survey of transfer learning. Journal of Big Data **3**(1) (2016) https://doi.org/10.1186/s40537-016-0043-6

[10] Nimma, D.D., Uddagiri, A.: Opt-stvit: Video recognition through optimized spatial-temporal video vision transformers. South Eastern European Journal of Public Health, 2103–2118 (2024) https://doi.org/10.70135/seejph.vi.2341

[11] Sowmyayani, S., Rani, P.A.J.: Content-based video retrieval system using two streams convolutional neural network. Multimedia Tools and Applications **82**(16), 24465–24483 (2023) https://doi.org/10.1007/s11042-023-14784-5

[12] Liu, X., Lee, J.-Y., Jin, H.: Learning video representations from correspondence proposals. In: 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp. 4268–4276. IEEE, (2019). https://doi.org/10.1109/cvpr.2019.00440 . http://dx.doi.org/10.1109/cvpr.2019.00440

[13] Yoon, H., Han, J.-H.: Content-based video retrieval with prototypes of deep features. IEEE Access **10**, 30730–30742 (2022) https://doi.org/10.1109/access.2022. 3160214

[14] KSII Transactions on Internet and Information Systems **15**(10) (2021) https://doi.org/10.3837/tiis.2021.10.011

[15] Tran, D., Bourdev, L., Fergus, R., Torresani, L., Paluri, M.: Learning spatiotemporal features with 3d convolutional networks. In: 2015 IEEE International Conference on Computer Vision (ICCV), pp. 4489–4497. IEEE, (2015). https://doi.org/10.1109/iccv.2015.510 . http://dx.doi.org/10.1109/iccv.2015.510

[16] Shalev, Y., Wolf, L.: Fine-Tuning CLIP via Explainability Map Propagation for Boosting Image and Video Retrieval, pp. 356–370. Springer, (2024). https://doi.org/10.1007/978-3-031-56027-9 22 . http://dx.doi.org/10.1007/978-3031-56027-9$_2$2

[17] Liu, Z., Ning, J., Cao, Y., Wei, Y., Zhang, Z., Lin, S., Hu, H.: Video swin transformer. In: 2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp. 3192–3201. IEEE, (2022). https://doi.org/10.1109/ cvpr52688.2022.00320 . http://dx.doi.org/10.1109/cvpr52688.2022.00320

[18] Tran, D., Wang, H., Torresani, L., Ray, J., LeCun, Y., Paluri, M.: A closer look at spatiotemporal convolutions for action recognition. In: 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition. IEEE, (2018). https://doi.org/10.1109/cvpr.2018.00675 . http://dx.doi.org/10.1109/cvpr.2018.00675

[19] Donahue, J., Hendricks, L.A., Guadarrama, S., Rohrbach, M., Venugopalan, S., Darrell, T., Saenko, K.: Long-term recurrent convolutional networks for visual recognition and description. In: 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 2625–2634. IEEE, (2015).

https://doi.org/ 10.1109/cvpr.2015.7298878 . http://dx.doi.org/10.1109/cvpr.2015.7298878

[20] Wang, Z., Yan, S., Zhang, X., Da Vitoria Lobo, N.: Self-supervised visual feature learning and classification framework: Based on contrastive learning. In: 2020 16th International Conference on Control, Automation, Robotics and Vision (ICARCV), pp. 719–725. IEEE, (2020). https://doi.org/10.1109/icarcv50220.2020.9305340 . http://dx.doi.org/10.1109/icarcv50220.2020.9305340

[21] Simonyan, K., Zisserman, A.: Two-Stream Convolutional Networks for Action Recognition in Videos (2014). https://arxiv.org/abs/1406.2199

[22] Wang, H., Schmid, C.: Action recognition with improved trajectories. In: 2013 IEEE International Conference on Computer Vision. IEEE, (2013). https: //doi.org/10.1109/iccv.2013.441 . http://dx.doi.org/10.1109/iccv.2013.441

[23] Chen, H., Hu, C., Lee, F., Lin, C., Yao, W., Chen, L., Chen, Q.: A supervised video hashing method based on a deep 3d convolutional neural network for large-scale video retrieval. Sensors **21**(9), 3094 (2021) https://doi.org/10.3390/s21093094

[24] Ullah, A., Muhammad, K., Hussain, T., Baik, S.W., De Albuquerque, V.H.C.: Event-oriented 3d convolutional features selection and hash codes generation using pca for video retrieval. IEEE Access **8**, 196529–196540 (2020) https://doi.org/10.1109/access.2020.3029834

[25] Kumar, V., Tripathi, V., Pant, B.: Learning unsupervised visual representations using 3d convolutional autoencoder with temporal contrastive modeling for video retrieval. International Journal of Mathematical, Engineering and Management Sciences **7**(2), 272–287 (2022) https://doi.org/10.33889/ijmems.2022.7.2.018

[26] Liu, Y., Wang, K., Lan, H., Lin, L.: Temporal Contrastive Graph Learning for Video Action Recognition and Retrieval (2021). https://arxiv.org/abs/2101. 00820

[27] Blondel, M., Fujino, A., Ueda, N.: Large-scale multiclass support vector machine training via Euclidean projection onto the simplex. In: 2014 22nd International Conference on Pattern Recognition, pp. 1289–1294. IEEE, (2014). https://doi.org/10.1109/icpr.2014.231 . http://dx.doi.org/10.1109/icpr.2014.231

[28] Soomro, K., Zamir, A.R., Shah, M.: UCF101: A Dataset of 101 Human Actions Classes from Videos in The Wild (2012). https://arxiv.org/abs/1212.0402

[29] Kuehne, H., Jhuang, H., Garrote, E., Poggio, T., Serre, T.: Hmdb: A large video database for human motion recognition. In: 2011 International Conference on Computer Vision, pp. 2556–2563 (2011). https://doi.org/10.1109/ICCV.2011. 6126543

[30] Iyer, R.R., Parekh, S., Mohandoss, V., Ramsurat, A., Raj, B., Singh, R.: Content-based Video Indexing and Retrieval Using Corr-LDA (2019). https://arxiv.org/ abs/1602.08581

[31] Hosna, A., Merry, E., Gyalmo, J., Alom, Z., Aung, Z., Azim, M.A.: Transfer learning: a friendly introduction. Journal of Big Data **9**(1) (2022) https://doi. org/10.1186/s40537-022-00652-w

[32] Banerjee, A., Kumar, E., Ravinder, M.: Learning clustered deep spatio-temporal prototypes using softmax regression for video information systems. International Journal of Information Technology **16**(5), 3085–3091 (2024) https://doi.org/10.1007/s41870-024-01826-w

[33] Banerjee, A., Kumar, E., Megavath, R.: Learning optimal deep prototypes for video retrieval systems with hybrid svm-softmax layer. International Journal of Data Science and Analytics (2024) https://doi.org/10.1007/s41060-024-00587-w