Learning Unsupervised Visual Representations using 3D Convolutional Autoencoder with Temporal Contrastive Modeling for Video Retrieval

Vidit Kumar
Department of Computer Science and Engineering,
Graphic Era Deemed to be University Dehradun, India.
Corresponding author: viditkumaruit@gmail.com

Vikas Tripathi
Department of Computer Science and Engineering,
Graphic Era Deemed to be University, Dehradun, India.
E-mail: vikastripathi.be@gmail.com

Bhaskar Pant
Department of Computer Science and Engineering,
Graphic Era Deemed to be University, Dehradun, India.
E-mail: pantbhaskar2@gmail.com

(Received on September 4, 2021; Accepted on January 25, 2022)

Abstract
The rapid growth of tag-free user-generated videos (on the Internet), surgical recorded videos, and surveillance videos has necessitated the need for effective content-based video retrieval systems. Earlier methods for video representations are based on hand-crafted, which hardly performed well on the video retrieval tasks. Subsequently, deep learning methods have successfully demonstrated their effectiveness in both image and video-related tasks, but at the cost of creating massively labeled datasets. Thus, the economic solution is to use freely available unlabeled web videos for representation learning. In this regard, most of the recently developed methods are based on solving a single pretext task using 2D or 3D convolutional network. However, this paper designs and studies a 3D convolutional autoencoder (3D-CAE) for video representation learning (since it does not require labels). Further, this paper proposes a new unsupervised video feature learning method based on joint learning of past and future prediction using 3D-CAE with temporal contrastive learning. The experiments are conducted on UCF-101 and HMDB-51 datasets, where the proposed approach achieves better retrieval performance than state-of-the-art. In the ablation study, the action recognition task is performed by fine-tuning the unsupervised pre-trained model where it outperforms other methods, which further confirms the superiority of our method in learning underlying features. Such an unsupervised representation learning approach could also benefit the medical domain, where it is expensive to create large label datasets.

Keywords- Contrastive learning, Convolutional autoencoder, Content-based search, Deep learning, Video retrieval, Future prediction, Unsupervised learning.

1. Introduction

Since the inception of the Internet, the number of videos produced, uploaded, and downloaded from the World Wide Web has been expanding constantly. Latest technologies have enabled users to upload video data to the internet (mostly social sites) recorded via cameras or smartphones. The YouTube statistics states that every minute, more than 2,000 hours of video are uploaded. Also, roughly 11 million videos are posted on Twitter daily. Most of these videos are unlabeled or semantic less tagged, making video analysis and searching a difficult task. These falsely semantically tagged clips or misrepresented short videos are also created to entice or mislead consumers by posing as fake news (Cao et al., 2020). In addition, other sources such as news agencies and surveillance networks have emerged in large quantities of video recording, requiring effective and efficient techniques for its management, indexing, and searching (Muhammad et al., 2021; Mühling et al., 2019; Subudhi et al., 2019). Furthermore, in the medical field, surgeries are often recorded in the hospital and stored due to law enforcement in some countries. Hence, there is the requirement for effective content-based analysis techniques for applications like surgeon skills assessment, for education purposes, surgical error analysis, etc (Kumar et al., 2021b). In addition, in the entertainment domain, watching several past movie scenes can be useful for the director to make an effective and original scene. With content-based search techniques, he/she can retrieve several other similar scenes in less time than manually searching for scenes in all previous movies (Deldjoo et al., 2018). Also, humans are prone to errors, and as a result, manual video search results could be erroneous and it is also a laborious and time-consuming task. Hence there is a need for very effective content-based analysis techniques. Importantly, it is required because most of these videos frequently lack semantic tags or labels. So, content-based video retrieval (CBVR) is a technique to video retrieval problems, that is, the problem of retrieving similar videos to a given query video. The term “content-based” means that the search is done by means of its visual features extracted from it (sometimes embedded audio may also be used). This requires an effective and robust spatiotemporal feature extraction method for video representation.

Earlier methods for video representation are built upon handcrafted features (Asha & Sreeraj, 2013; Araujo et al., 2017; Brindha & Visalakshi, 2017; Ram et al., 2020; Zhu et al., 2016), which are not good at effectively representing video dynamics. Later, deep learning has emerged as successful and powerful in computer vision tasks that include classification (Karpathy et al., 2014; Krizhevsky et al., 2012), segmentation (Shelhamer et al., 2017), gesture recognition (Jain et al., 2020a, 2020b), object detection (Ren et al., 2016) and retrieval (Babenko et al., 2014). The key to this success is the use of massively labeled data and effective deep learning models. However, collecting a labeled dataset of videos on such a large scale can be costly. Therefore, taking advantage of unlabeled videos, which are freely available on the Internet, can be an economical alternative. Hence, the big challenge in artificial intelligence is to teach the computer useful representations without any supervision. Unsupervised learning is also essential for the tasks like anomaly detection (Pang et al., 2021), surgical video analysis (Paysan et al., 2021; Wu et al., 2021) etc., due to the lack of labeled data. As for unsupervised learning of video representations, a lot of work has been proposed in this direction in recent times which is based on self-supervised learning. However, most of these methods are built over a single predefined pretext task (Benaim et al., 2020; Cho et al., 2021; Jing et al., 2018; Kim et al., 2019; Wang et al., 2020), which usually transforms video and train the network to predict the transformation. Instead, this research investigates 3D convolutional autoencoder (3D-CAE) as an unsupervised learning approach to learn video representations. In addition, this paper also explores a contrastive learning approach to further enhance representation learning.

The main contributions of this research work are as follows:

- This paper explores a 3D-CAE network in learning video representations for the task of video retrieval.
- In addition, a new unsupervised video representation learning method is proposed that learns representations by joint learning of past and future prediction, constraint by temporal coherence aware contrastive loss.
- For smooth training and efficient learning, a 3D-CAE network based on a C3D network has been designed.

- And finally, the proposed approach was tested on the UCF-101 and HMDB-51 datasets, where the proposed approach yields better results compared to state-of-the-art.

The remaining part of this work is divided as: related works are discussed in Section 2, where we explored previous studies for representing videos with a current research focus. In Section 3, the proposed method is elaborated, where the components of the proposed work along with the backbone neural network are discussed first and then the proposed method. Then, experiments with results are discussed in Section 4, where two datasets are chosen to validate the proposed approach. And finally, Section 5 concludes the paper.

2. Related Work

Since the 1990s (Rui et al., 1988), considerable research has been done in image retrieval (Zhou et al., 2017), but CBVR has been ignored in the multimedia community. Due to recent advancements in technologies, there is a need for CBVR. Most of the earlier methods rely on hand-designed descriptors. For example, in Jiang et al. (2007) and Wang et al. (2012), bag-of-words model-based methods are proposed for video representation. In Asha and Sreeraj (2013), the SURF features-based video retrieval method is proposed. In addition, SIFT-based methods are also proposed in Zhu et al. (2016). Additionally, Brindha and Visalakshi (2017) combined low-level features such as texture, colour and shape with a higher-level representation such as a motion to reduce the gap between them. In Araujo et al. (2017), the bloom filter-based query image video retrieval method is proposed. Ram et al. (2020) proposed the histogram-based similarity measure for the collation of HOG descriptor encoded video frames. The main limitation of all methods is that, while these features are useful for representing video at the static frame level, they are ineffective at representing video sequences. Furthermore, since these methods are designed based on hand-crafted features, any further improvements will depend on effective feature engineering.

Since CNNs have recently seen tremendous success in the computer vision field, particularly with the task of image classification, object detection, segmentation, and retrieval, CNNs are rapidly increasing in popularity. This has also led to research into video and image retrieval. For example, the CNN-based work is proposed by Babenko et al. (2014) where they investigated the deep descriptors for image search. Further, a fine-grain image retrieval method is proposed in Kumar et al. (2020), where they proposed CNN based framework for fine-grained image search. This progress also resulted in video retrieval, for instance, CNN features-based video retrieval method is proposed in Podlesnaya and Podlesnyy (2016) and Lou et al. (2017). In Markatopoulou et al. (2017) multiple pre-trained CNN models are first used to detect key objects and scenes in video frames, and complex linguistic rules are designed for video representation. Moreover, in Ueki et al. (2017) concept-based method is proposed, where they extract pre-trained CNN features from each frame and then perform element-wise max pooling. Then they used SVM to build concept scores to use for searching. In addition, Mühling et al. (2017, 2019) and Kumar et al. (2022) also exploited CNN features for video retrieval problems. In the direction of video classification, more specifically action recognition task, approaches such as Simonyan and Zisserman (2014), Karpathy et al. (2014) used 2D CNN to represent and analyze video contents. Further, this is improved by exploring 3D CNN (Tran et al., 2015, 2018). However, all the above methods are based on models that are trained on a large-scale labeled dataset of images or videos. Any further improvement in recognition or retrieval accuracy will require training on either a large-scale dataset or effective network designing. However, collecting a large-scale labeled video dataset is an expensive task compared to a labeled dataset of images. Thus, the economical alternative is to learn video representations using freely available web videos, which is recently emerged as a major research site in computer vision. In this regard, most of the work recently proposed is based on self-supervised learning.

Self-supervised learning is a method of learning representations without using labels, where a task is usually generated as a "pretext task" by some transformation function applied to the dataset. For example, in Misra et al. (2016), a frames order verification task using a CNN as a backbone is proposed. In Lee et al. (2017), a frames sorting pretext task is designed to learn self-supervised features. Moreover, Fernando et al. (2017) proposed an odd-one-function. In Buchler et al. (2018), the author designed a sample permutation policy and applied deep reinforcement learning for frames order prediction problems. Most of these techniques used 2D CNN, which seems to be lacking when it comes to motion capture. In order to deal with this, a combination of 3D CNN and pretext tasks can be the most effective option. In this direction, Jing et al. (2018) proposed a 3D CNN-based feature learning approach by utilizing a video rotation prediction task. In addition, using 3D CNN, a space-time Cubik puzzle-solving task is proposed for video representation learning in Kim et al. (2019). Also, Xu et al. (2019) proposed 3D CNN-based clip order prediction task by extending the frame order prediction task (Lee et al., 2017). Further, some work based on speed-based pretext task (Wang et al., 2020; Benaim et al., 2020; Cho et al., 2021) was proposed, which also uses 3D CNN as a backbone network. When it comes to learning representations using unlabeled data, it is clear that the usefulness of a learned representation for downstream tasks depends on the pretext task.

In contrast, this paper explores a 3D convolutional auto-encoder to learn video representations. In addition, a multi-pretext task is designed to predict future frames and past frames using 3D-CAE. In Table 1, we highlighted the key references related to the proposed work.

Table 1. Key related papers and their comparison.

Author	Network	Methodology (Self-Supervised Pretext Task)	Limitation
Noroozi and Favaro (2016)	Alexnet	Prediction of shuffled space Jigsaw puzzles	• Only applicable to a CNN that accepts one or two frames as input • Not suitable for Spatio-temporal representation.
Lee et al. (2017)		Sorting the Sequences	
Buchler et al. (2018)		Reinforcement learning + sorting sequences	
Benaim et al. (2020)	S3D-G	Prediction of the speed of video play	• Applicable to a 3DCNN that accepts 32 or 64 frames as input • Used Single pretext task
Jing et al. (2018)	R18-3D	Prediction of video rotation	• Based on Single Spatial transformation pretext task. • Avoids temporal information.
Luo et al. (2020)	C3D	Video cloze procedure	• Based on a Single pretext task
Xu et al. (2019)		Prediction of Video clip order	
Proposed approach		**Multi-pretext task (Future prediction + Past prediction + Temporal coherence aware contrastive learning)**	• **Limitation can be in computations involving in 3DCNN**

3. Method

Consider the set $V = \left[V_1, V_2, ..., V_q\right]$ of q unlabeled training videos and $V \in R^{M\times N\times C\times F}$ where F denotes the number of frames and $M\times N\times C$ is frame's spatial resolution size in which C refers to the color channel. The objective is to learn visual representations of videos V using an unsupervised learning approach. In this regard, a 3D-CAE is designed and explores its power in learning video representations. In addition, a multitask learning method based on 3D-CAE is proposed to strengthen the representation. An overview of the proposed method is depicted in Figure 1, where the multitask objective in the form of past and future frame prediction is jointly implemented with a temporal contrastive learning module on the top of the shared encoder. Next, we discuss the vanilla autoencoder and convolutional autoencoder first and then 3D-CAE with the proposed approach.

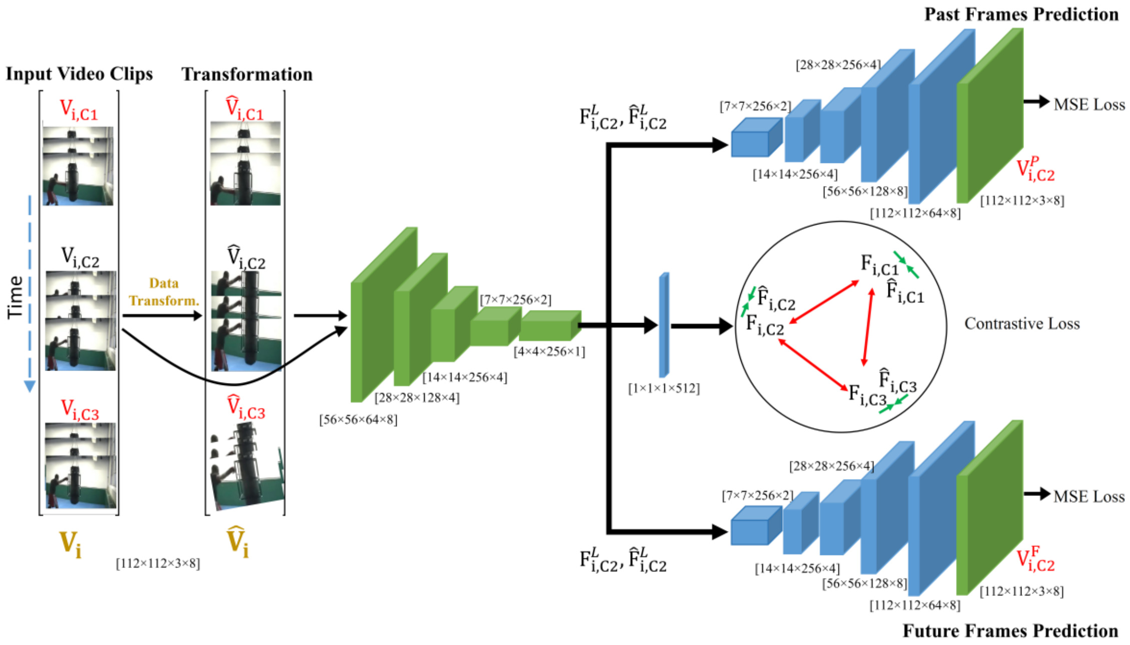


Figure 1. Overview of proposed Approach. (The red arrow indicates repulsion and the green arrow indicates attraction.)

3.1 Preliminaries

3.1.1 Autoencoder

An autoencoder is a special type of feed-forward network consisting of an encoder-decoder architecture. The idea of autoencoder was first mentioned in Rumelhart et al. (1985) to use backpropagation without supervision. It is then used in training deeper networks like in Hinton et al. (2006) and Bengio et al. (2007). The basic autoencoder consists of 3 layers: input layer, hidden layer and output layer as shown in Figure 2 (a). All the connections between layers of encoder and decoder are fully connected i.e. from the previous layer, all the units are connected to the following layer. The objective of this network is to learn low-dimensional encoding in an unsupervised way i.e. without requiring input labels from the system. The network uses a back-propagation algorithm for training. Autoencoders are generally used as dimensionality reduction technique because it learns low embedding which can be used as a feature representation.

The general autoencoder involves two process: encoding and decoding. Consider the input $x \in R^n$, in encoding phase, the hidden layer h is obtain by the encoding function $h = En(x)$. The encoding function En(x) is defined as:

$$h = En(x) = \delta(W_E x + b) \tag{1}$$

Where, $W_E \in R^{m \times n}$ is weight matrix, $b \in R^m$ is bias and δ is the activation function.

In decoding phase, the output layer y is obtain by the decoding function $y = De(h)$. The decoding function De(x) is defined as:

$$y = De(h) = \delta(W_D h + \hat{b}) \tag{2}$$

Where, $W_D \in R^{n \times m}$ is weight matrix, $\hat{b} \in R^n$ is bias.

Then the autoencoder is trained to make output y to be as close as possible to input x. This is achieved by minimizing mse loss function (3), which is typically used in these networks.

$$L(x, y) = \|x - y\|^2 \tag{3}$$

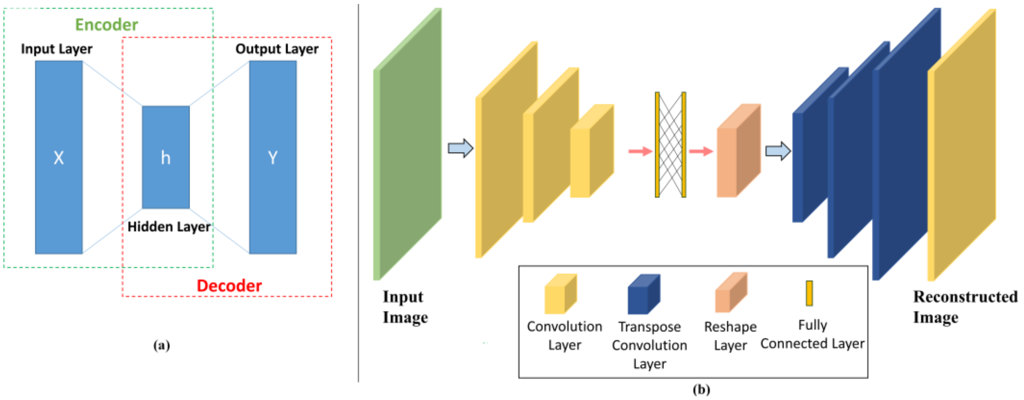


Figure 2. (a) General Autoencoder architecture. (b) General Convolutional Autoencoder architecture.

3.1.2 Convolutional Autoencoder (2D-CAE)

Image features can also learn via vanilla autoencoder. However vanilla autoencoder learn features globally since it is fully connected in nature. Because of this, important local features in the images are lost. The solution to this problem is to take advantage of convolutional layers in both the encoder and the decoder to extract local features. In addition, multiple convolutional layers can be added to the encoder to encode different types of features. 2D-CAEs, thus, have convolutional encoding layers and transpose convolutional decoding layers. The convolutional layers are for extracting local image features while maintaining its spatial structure whereas transpose convolutional layers are for restoring spatial structure back to original form. The general architecture is shown in Figure 2 (b). For reducing spatial structure in encoder and for upscaling spatial structure in decoder can be done by employing maxpooling and unpooling layers (where unpooling operation is done by using the locations stored during the maxpooling operation). However, this can also be achieved by the stride factor in convolutional kernels when performing the convolution operation. The encoding operation can be computed as:

$$h_c = En_c(I) = \delta_c(I * W_{CE} + b') \tag{4}$$

Where, I is a matrix (image), $W_{CE} \in R^{d\times d}$ is a kernel (weighted filter), b' is bias and δ_c is activation function (commonly used is RELU) and $*$ denotes convolutional operation.

Similarly, the decoding operation can be computed as:

$$I' = De_c\left(h_c\right) = \delta_c\left(\left(En_c(I)\right) * W_{CD} + b''\right) \tag{5}$$

Where, I' is a reconstructed matrix (image), $W_{CD} \in R^{d\times d}$ is a kernel (weighted filter), b'' is bias.

The loss to minimize is given as

$$L(I, I') = \left\|I - I'\right\|^2 \tag{6}$$

Table 2. 3D-CAE network configuration.

	Layer	Output Size
Encoder	Input	112×112×3×8
	Conv-1	112×112×64×8
	MaxPool-1	56×56×64×8
	Conv-2	56×56×128×8
	MaxPool-2	28×28×128×4
	Conv-3	28×28×256×4
	MaxPool-3	14×14×256×4
	Conv-4	14×14×256×4
	MaxPool-4	7×7×256×2
	Conv-5	7×7×256×2
	MaxPool-5	4×4×256×1
Decoder	Tconv-1	7×7×256×2
	Tconv-2	14×14×256×4
	Tconv-3	28×28×256×4
	Tconv-4	56×56×128×8
	Tconv-5	112×112×64×8
	Conv-F	112×112×3×8

3.2 Network Architecture

The 2D-CAE is generally good at capturing visual information. However, the most important information in video is motion which cannot be captured by 2D-CAE. Recently, 3D convolution based networks (Tran et al., 2015, 2018) have been shown to be effective in capturing temporal dynamics in the video. Therefore, to learn Spatio-temporal features, we designed a 3D-CAE following C3D architecture (Tran et al., 2015). The architecture details of 3D-CAE is depicted in Table 2. We follow the small variant of C3D to design the encoder. Instead of 16 frames, a clip of 8 frames (skipped every second frame) is sampled from the video and input to the network. The convolutional kernel size is set to 3×3×3 throughout the network. The convolution operation is performed with 1×1×1 stride and the transpose convolution is performed according to the required stride. For downsampling the spatial resolution, max pooling is deployed after each convolutional layer. In total, the encoder contains 5 convolutional layers and 5 maxpooling layers. Whereas, the decoder employs the encoder's inverse structure, substituting transposed convolutional layers instead of Conv-pool blocks. Appropriate zero-padding is applied to the convolution and max pooling layer (where applicable) to ensure that the dimensions of all inputs and outputs of the convolution operation are consistent.

3.3 Multi-task Learning (MTL) based on 3D-CAE

MTL has been found to generate more efficient models than single-task learning (Caruana, 1997). MTL aims to improve learning efficiency and accuracy by concurrently optimizing multiple objectives while using shared representations (Huang et al., 2014; Jian et al., 2020a, 2000b; Yao et al., 2020). In line with this idea, to further improve the video representation power, we propose to explore past and future frames prediction pretext tasks using 3D-CAE in a multi-task learning setting.

Let $V_{i,Cj}$ denotes the j^th video clip sampled from the i^th video of the training set, where j = {1, 2, 3} (Here, we consider the three clips representing current $V_{i,C2}$, past $V_{i,C1}$ and future $V_{i,C3}$ clips sampled from random locations in the video.) For past frames prediction (PP) task, the loss can be given as:

$$L_P(V_{i,C1}, V_{i,C2}^P) = \left\| V_{i,C1} - V_{i,C2}^P \right\|^2 \tag{7}$$

Where, $V_{i,C2}^P$ is the generated past video frames.

Similarly, for future frames prediction (FP) task, the loss can be given as:

$$L_F(V_{i,C3}, V_{i,C2}^F) = \left\| V_{i,C3} - V_{i,C2}^F \right\|^2 \tag{8}$$

Where, $V_{i,C2}^F$ is the generated future video frames.

To further regularize the learning process, we also propose to leverage contrastive learning (Chen et al., 2020) as an additional objective. Generally, the purpose of contrastive learning is to learn representations by separating positive pairs from the negative ones in the latent space. As in Chen et al. (2020), different transformations of same sample are considered positives while transformations of other samples are treated as negatives. We extend this idea to the video domain by exploiting temporal coherence in the video. Since the video clips at different time steps of the same long video have different temporal information, their representations should be different. Thus, clips sampled from different time steps of the same video can be considered as negatives while positives can be created by data transformation.

Let $Tr(.)$ be the transformation function (such as rotation, cropping, flipping) which transform $V_{i,Cj}$ to $\widehat{V}_{i,Cj}$. The positive pairs formed are $\left\{\left(V_{i,Cj}, \widehat{V}_{i,Cj}\right)\right\}$ and the negative pairs are $\left\{\left(V_{i,Cj}, V_{i,Ck}\right)\right\}, j \neq k$. Let $F_{i,Cj}$ and $\widehat{F}_{i,Cj}$ be the L2 normalized embeddings of $V_{i,Cj}$ and $\widehat{V}_{i,Cj}$ encoded by the deep network (here, encoder of 3D-CAE followed by fully-connected layer of 512 units; see Figure 1). Then positive pairs of feature vector are $\left\{\left(F_{i,Cj}, \widehat{F}_{i,Cj}\right)\right\}$ and negative pairs are $\left\{\left(F_{i,Cj}, F_{i,Ck}\right)\right\}, j \neq k$. The objective is to make positive pairs $\left\{\left(F_{i,Cj}, \widehat{F}_{i,Cj}\right)\right\}$ closer and negative pairs farther apart in the embedding space. In other words, in terms of cosine similarity (CosSim), the constraint to be apply is $\operatorname{Cos}\operatorname{Sim}\left(F_{i,Cj}, \widehat{F}_{i,Cj}\right) > \operatorname{Cos}\operatorname{Sim}\left(F_{i,Cj}, F_{i,Ck}\right), k \neq j$, where $\operatorname{Cos}\operatorname{Sim}\left(F_{i,Cj}, F_{i,Ck}\right)$ is $F_{i,Cj}{}^{T} \bullet F_{i,Ck}$ which is the dot product between two feature vectors.

Given a K minibatch of videos, the temporal coherence aware contrastive learning loss (TCCL) can be given as:

$$L_{TCCL} = \frac{1}{|K|}\sum_{i}\sum_{j}\left(-\log\frac{\exp\left(\operatorname{Cos}\operatorname{Sim}\left(F_{i,Cj}, \widehat{F}_{i,Cj}\right)/\tau\right)}{\sum_{k}\exp\left(\operatorname{Cos}\operatorname{Sim}\left(F_{i,Cj}, \widehat{F}_{i,Ck}\right)/\tau\right) + \sum_{k\neq j}\exp\left(\operatorname{Cos}\operatorname{Sim}\left(F_{i,Cj}, F_{i,Ck}\right)/\tau\right)}\right) \tag{9}$$

The total loss for joint learning of multi-tasks is:

$$L = L_{P} + \lambda_{1}L_{F} + \lambda_{2}L_{TCCL} \tag{10}$$

3.4 Implementation Details

We implement our proposed approach in MATLAB 2019b with NVIDIA tesla k40c GPU enabled Xeon E5 system. During training, K videos are sampled from the training set. Then 3 clips are extracted from each video from different temporal locations. Each video clip is resized to 128×170×3×8, then 112×112×3×8 crop is randomly sampled from it. After that, each clip undergoes data transformation, and both the augmented and original clips are fed to the network for feature learning. The minibatch size is set to $K = K_{1}*3*2 = 30$. For data transformation, we use random cropping, rotation, channel replication and horizontal flipping. Adam optimization is used and the initial learning rate is set to 1e-3 and decrease by 1/10 every 7k iterations. The embedding size is chosen to 512 neurons for a contrastive learning tasks, i.e. a fully connected layer (512 units) is added on top of the encoder part of the 3D-CAE. The implementation details are summarized in Table 3.

Table 3. Implementation details.

Tool Used	MATLAB 2019b
GPU Used	NVIDIA Tesla k40c GPU
CPU Used	Xeon E5 system
Minibatch Size (K)	30
Sampled Video Size	128×170×3×8
Input Video Size	112×112×3×8
Learning Rate	1e-3 decrease by 1/10 every 7k iterations
Optimizer	Adam
Embedding Size For Contrastive Learning	512
Data Transformation	Random Cropping, Random Rotation, Channel Replication and Horizontal Flipping

4. Experiments

4.1 Dataset and Evaluation Setting

UCF-101 (Soomro et al., 2012) and HMDB-51 (Kuehne et al., 2011) datasets are selected to conduct the experiments. UCF-101 consists of 13k video clips sampled from various sources such as sports videos, daily lifelog videos, etc. It is divided into 101 human actions. HMDB-51 is another dataset consisting of 7k video clips sampled from different movies and YouTube. It is divided into 51 human actions. For unsupervised pre-training, the UCF-101 training split-1 without labels is used. And, then testing is done for both the datasets such that the testing split-1 is considered as query set and training split-1 as retrieval set.

This research follows the setting of Xu et al. (2019) for evaluation of retrieval performance, where 10 clips per video are sampled and centered cropped. Pool5's activations are used to represent each clip. Cosine distance is used for video clip similarity measurement and top-k retrieval accuracy is computed such that a query clip is said to be correctly predicted if the label of the query clip is found in the top-k clips retrieved from the training split-1.

4.2 Results

4.2.1 Influence of Joint Learning of Multi-task on Retrieval Performance

First, we see the effect of joint learning in our method on video retrieval performance. We perform this experiment on the UCF-101 dataset, and the results are reported in Table 4. We can see that video representations learned through 3D-CAE are capable of performing well compared to random initialization features. With future frames prediction task and past frames prediction task, the features learned on top of 3D-CAE show further improvement which is reflected in retrieval accuracy. Then jointly learning of future frames prediction task and past frames prediction task with temporal coherence aware contrastive learning (FP + PP + TCCL) further improves the retrieval performance.

Table 4. Impact of multi-task learning on retrieval performance (Clip Level).

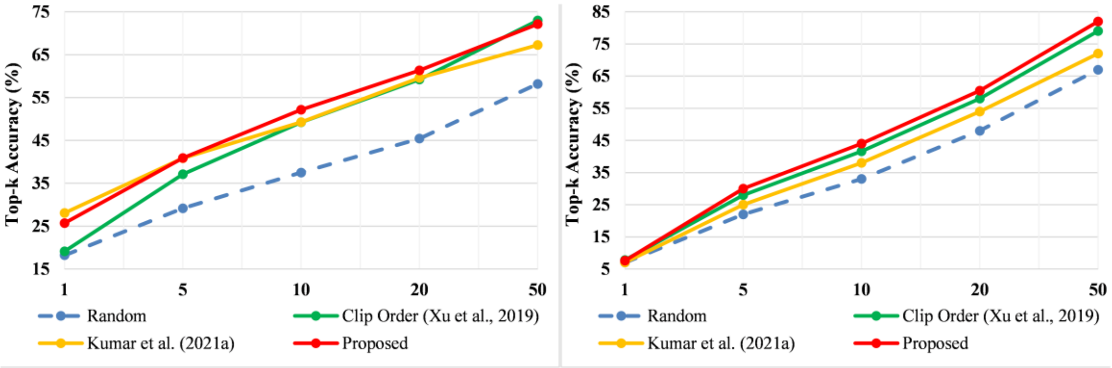
Methods	**k=1**	**k=5**	**k=10**	**k=20**	**k=50**
Random	20.23	27.26	31.21	36.17	44.46
3D-CAE	22.87	31.04	36.15	42.37	51.48
Future Prediction (FP)	24.47	33.11	38.24	44.41	54.93
Past Prediction (PP)	24.41	32.26	37.92	44.69	55.13
FP + PP	24.52	33.18	38.65	45.73	56.98
FP + PP + TCCL	**28.13**	**37.15**	**44.09**	**53.78**	**66.44**

Table 5. (Clip Level) Top-k retrieval accuracy (%).

Network	**Methods**	**UCF-101**					**HMDB-51**				
		k=1	**k=5**	**k=10**	**k=20**	**k=50**	**k=1**	**k=5**	**k=10**	**k=20**	**k=50**
C3D	Random	20.23	27.26	31.21	36.17	44.46	7.2	13.5	21.9	30.65	41.72
Alexnet	Jigsaw (Noroozi & Favaro, 2016)	19.7	28.5	33.5	40.0	49.4	-	-	-	-	-
	OPN (Lee et al., 2017)	19.9	28.7	34.0	40.6	51.6	-	-	-	-	-
	Buchler et al. (2018)	25.7	36.2	42.2	49.2	59.5	-	-	-	-	-
S3D-G	SpeedNet (Benaim et al., 2020)	13.0	28.1	37.5	49.5	65.0	-	-	-	-	-
C3D	VCP (Luo et al., 2020)	17.3	31.5	42.0	52.6	67.7	7.4	22.6	34.4	48.5	70.1
	Clip Order (Xu et al., 2019)	12.5	29.0	39.0	50.6	66.9	7.8	23.8	35.5	49.3	71.6
	Kumar et al. (2021a)	28.17	37.92	43.24	51.41	62.93	7.5	21.5	32.62	46.23	68.19
	Proposed (FP + PP + TCCL)	**28.13**	**37.15**	**44.09**	**53.78**	**66.44**	**7.6**	**24.23**	**35.62**	**51.42**	**72.18**

4.2.2 Comparison to State-of-the-arts

In Table 5, we compare our approach with other methods, where we can see that our approach able to outperform other methods on both datasets. On UCF-101, the proposed method achieve 28.13% top-1, 37.15% top-5, 44.09% top-10, 53.78% top-20 and 66.44% top-50 retrieval accuracy. On closer analysis, we can see that the proposed method has advantages over 2D-CNN based methods and also outperforms 3D-CNN based methods like Xu et al. (2019) and Luo et al. (2020). It is also slightly better than the recent method of Kumar et al. (2021a). On HMDB-51, the proposed method also performs well with 24.23% top-5, 35.62% top-10, 51.42% top-20 and 72.18% top-50 retrieval accuracy. This further confirms with multi-task learning, the network learns more generic features. We also evaluate the retrieval performance at the video level. For this, we average the features of the 10 extracted clips to represent the video. As depicted in Figure 3, with the proposed method, the video level retrieval accuracy is increased compared to that of the randomly initialized network and other methods.



(a) UCF-101

(b) HMDB-51

Figure 3. Video retrieval results (top-k accuracy %) on UCF-101 and HMDB-51.

4.2.3 Visualization

In Figure 4, using the method of Zagoruyko and Komodakis (2016) we visualize the activations of conv 5 layers of the pre-trained model, where we can see the model learned to pay attention to the object motion's areas. Here we used 5 random videos sampled from the UCF-101 for attention visualization. And it can be seen that the neurons are highly active on the object motions region which is highlighted with red.

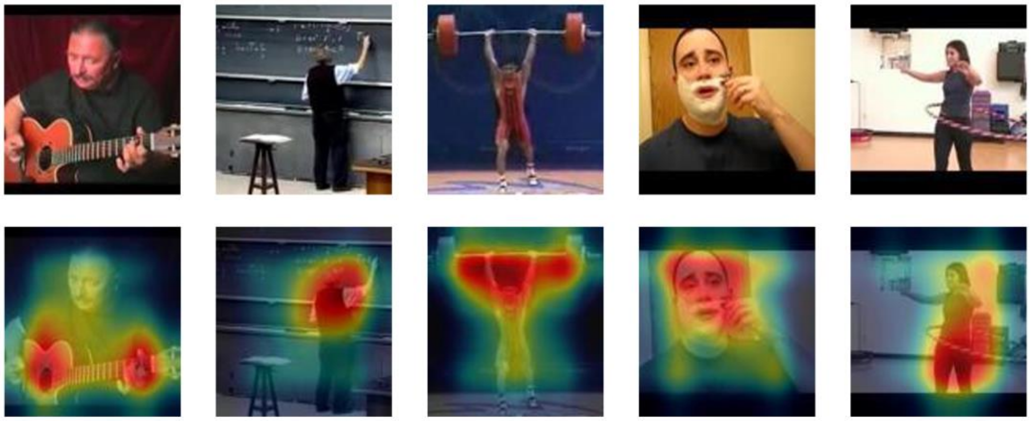


Figure 4. Attention visualization on random video samples (UCF-101) based on an unsupervised pre-trained model. (For each column, the first row represents input video and the second row represents its neural activations.)

4.2.4 Ablation Study (Action Recognition)

We also conduct an ablation study in form of an action recognition task, where we want to see if convolutional kernels' weight initialized with unsupervised pre-training improves action recognition task compared to randomly initialized weights. Following the action recognition evaluation protocol (Xu et al., 2019), we fine-tuned the model and test the accuracy on both datasets. The results are reported in Table 6 along with a comparison to other methods, where we achieve 67.62% on UCF-101 and 31.22% on HMDB-51. This further confirms that with the proposed approach the network is able to learn the initial level features which are reflected in the initialization of the network. Compared to other methods, the proposed approach is able to outperform others on UCF-101, and most of the methods on HMDB-51. The methods Benaim et al. (2020) and Jing et al. (2018) achieves better results on HMDB51 than the proposal. The reason is that they trained the network on much larger dataset and also utilized heavy 3D-CNN networks than this work.

Table 6. Action recognition results.

Method	Network	UCF-101	HMDB-51
Jigsaw (Noroozi & Favaro, 2016)	Alexnet	51.5	-
Shuffle&Learn (Misra et al., 2016)	Alexnet	50.9	-
OPN (Lee et al., 2017)	Alexnet	56.3	-
Buchler et al. (2018)	Alexnet	58.6	-
CubicPuzzle (Kim et al., 2019)	C3D	60.6	28.3
3D RotNet (Jing et al., 2018)	3D-ResNet18	66.0	37.1
Speednet (Benaim et al., 2020)	I3D	66.7	43.7
Clip Order (Xu et al., 2019)	C3D	65.6	28.4
Kumar et al. (2021a)	C3D	66.8	25.6
Proposed (FP + PP + TCCL)	**C3D**	**67.62**	**31.22**

5. Conclusion

In this paper, a novel unsupervised video representation learning technique is proposed, where video features are learned via joint learning of future frames and past frames prediction pretext task. This learning further regularized by temporal coherence aware contrastive learning, which helps the model to learn more generic features. For the future frames and past frames prediction pretext task, the 3D-CAE network is designed based on C3D network structure. The learned features are validated on the UCF-101 and HMDB-51 datasets, where the proposed approach outperforms previous methods on both datasets. An ablation study was also conducted with respect of action recognition task, where better accuracy was achieved on both datasets compared to state-of-the-arts. The limitation of our approach may be computation, as it is based on 3D-CNN and it is costlier than 2D-CNN. For future work, further research is needed to explore more self-supervised learning methods as well as exploration of deep and efficient networks for video representation learning, and also to explore contrastive learning as it is showing as a strong self-supervised learning approach. In addition, future work also includes testing and developing new methods for other domains, like medical, where creating labeled dataset is expensive.

Conflict of Interest

The authors confirm that there is no conflict of interest to declare for this publication.

Acknowledgments

This research did not receive any specific grant from funding agencies in the public, commercial, or not-for-profit sectors.

References

Araujo, A., & Girod, B. (2017). Large-scale video retrieval using image queries. *IEEE Transactions on Circuits and Systems for Video Technology*, *28*(6), 1406–1420. doi: https://doi.org/10.1109/TCSVT.2017.2667710.

Asha, S., & Sreeraj, M. (2013, August). Content-based video retrieval using SURF descriptor. In *2013 Third International Conference on Advances in Computing and Communications* (pp. 212–215). India: IEEE.

Babenko, A., Slesarev, A., Chigorin, A., & Lempitsky, V. (2014, September). Neural codes for image retrieval. In *European Conference on Computer Vision* (pp. 584–599). Cham, Zurich, Switzerland: Springer.. doi: https://doi.org/10.1007/978-3-319-10590-1_38.

Benaim, S., Ephrat, A., Lang, O., Mosseri, I., Freeman, W. T., ... & Dekel, T. (2020). Speednet: Learning the speediness in videos. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (pp. 9922–9931). IEEE. doi: https://doi.org/10.1109/CVPR42600.2020.00994.

Bengio, Y., Lamblin, P., Popovici, D., & Larochelle, H. (2007). Greedy layer-wise training of deep networks. In *Advances in neural information processing systems* (pp. 153–160). Canada: MIT Press.

Brindha, N., & Visalakshi, P. (2017). Bridging semantic gap between high-level and low-level features in content-based video retrieval using multi-stage ESN–SVM classifier. *Sādhanā*, *42*(1), 1–10.

Buchler, U., Brattoli, B., & Ommer, B. (2018). Improving spatiotemporal self-supervision by deep reinforcement learning. In *Proceedings of the European Conference on Computer Vision (ECCV)* (pp. 770–786). Cham, Munich, Germany: Springer. doi: https://doi.org/10.1007/978-3-030-01267-0_47

Cao, J., Qi, P., Sheng, Q., Yang, T., Guo, J., & Li, J. (2020). Exploring the role of visual content in fake news detection. In K. Shu, S. Wang, D. Lee, & H. Liu (eds.), *Disinformation, misinformation, and fake news in social media. Lecture notes in social networks* (pp. 141–161). Chem: Springer.

Caruana, R. (1997). Multitask learning. *Machine Learning*, *28*(1), 41–75.

Chen, T., Kornblith, S., Norouzi, M., & Hinton, G. (2020, November). A simple framework for contrastive learning of visual representations. In *International Conference on Machine Learning* (pp. 1597–1607). PMLR.

Cho, H., Kim, T., Chang, H. J., & Hwang, W. (2021). Self-supervised Visual Learning by variable playback speeds prediction of a video. *IEEE Access*, 9, 79562–79571.

Deldjoo, Y., Constantin, M. G., Ionescu, B., Schedl, M., & Cremonesi, P. (2018, June). MMTF-14K: A multifaceted movie trailer feature dataset for recommendation and retrieval. In *Proceedings of the 9th ACM Multimedia Systems Conference* (pp. 450–455). doi: https://doi.org/10.1145/3204949.3208141.

Fernando, B., Bilen, H., Gavves, E., & Gould, S. (2017). Self-supervised video representation learning with odd-one-out networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (pp. 3636–3645). Honolulu, HI: IEEE. doi: https://doi.org/10.1109/CVPR.2017.607.

Hinton, G. E., Osindero, S., & Teh, Y. W. (2006). A fast learning algorithm for deep belief nets. *Neural Computation*, *18*(7), 1527–1554. doi: https://doi.org/10.1162/neco.2006.18.7.1527.

Huang, W., Song, G., Hong, H., & Xie, K. (2014). Deep architecture for traffic flow prediction: Deep belief networks with multitask learning. *IEEE Transactions on Intelligent Transportation Systems*, *15*(5), 2191–2201. doi: https://doi.org/10.1109/TITS.2014.2311123.

Jain, D. K., Mahanti, A., Shamsolmoali, P., & Manikandan, R. (2020a). Deep neural learning techniques with long short-term memory for gesture recognition. *Neural Computing and Applications*, *32*(20), 16073–16089. doi: https://doi.org/10.1007/s00521-020-04742-9.

Jian, Z., Yue, W., Wu, Q., Li, W., Wang, Z., & Lam, V. (2020b, November). Multitask learning for video-based surgical skill assessment. In *2020 Digital Image Computing: Techniques and Applications (DICTA)* (pp. 1–8). Melbourne, Australia: IEEE.. doi: https://doi.org/10.1109/DICTA51227.2020.9363408.

Jiang, Y. G., Ngo, C. W., & Yang, J. (2007, July). Towards optimal bag-of-features for object categorization and semantic video retrieval. In *Proceedings of the 6th ACM International Conference on Image and Video Retrieval* (pp. 494–501). Amsterdam, The Netherlands: ACM..

Jing, L., Yang, X., Liu, J., & Tian, Y. (2018). Self-supervised spatiotemporal feature learning via video rotation prediction. *arXiv preprint arXiv:1811.11387*.

Karpathy, A., Toderici, G., Shetty, S., Leung, T., Sukthankar, R., & Fei-Fei, L. (2014). Large-scale video classification with convolutional neural networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (pp. 1725–1732). Columbus, OH: IEEE.

Kim, D., Cho, D., & Kweon, I. S. (2019, July). Self-supervised video representation learning with space-time cubic puzzles. *Proceedings of the AAAI Conference on Artificial Intelligence, 33*(01), 8545–8552.

Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2012). Imagenet classification with deep convolutional neural networks. In F. Pereira, C. J. C. Burges, L. Bottou, & K. Q. Weinberger (eds.), *Advances in neural information processing systems* (Vol. 25, pp. 1097–1105). Curran Associates, Inc.

Kuehne, H., Jhuang, H., Garrote, E., Poggio, T., & Serre, T. (2011, November). HMDB: A large video database for human motion recognition. In *2011 International Conference on Computer Vision* (pp. 2556–2563). Barcelona, Spain: IEEE. doi: https://doi.org/10.1109/ICCV.2011.6126543.

Kumar, V., Tripathi, V., & Pant, B. (2020, February). Content based fine-grained image retrieval using convolutional neural network. In *2020 7th International Conference on Signal Processing and Integrated Networks (SPIN)* (pp. 1120–1125). Noida, India: IEEE.

Kumar, V., Tripathi, V., & Pant, B. (2021a, April). Unsupervised learning of visual representations via rotation and future frame prediction for video retrieval. In *International Conference on Advances in Computing and Data Sciences* (pp. 701–710). Cham: Springer.

Kumar, V., Tripathi, V., & Pant, B. (2021b, July). Content based surgical video retrieval via multi-deep features fusion. In *2021 IEEE International Conference on Electronics, Computing and Communication Technologies (CONECCT)* (pp. 1–5). Bangalore, India: IEEE. doi: https://doi.org/10.1109/CONECCT52877.2021.9622562.

Kumar, V., Tripathi, V., & Pant, B. (2022). Exploring the strengths of neural codes for video retrieval. In *Machine learning, advances in computing, renewable energy and communication* (pp. 519–531). Springer, Singapore. doi: https://doi.org/10.1007/978-981-16-2354-7_46.

Lee, H. Y., Huang, J. B., Singh, M., & Yang, M. H. (2017). Unsupervised representation learning by sorting sequences. In *Proceedings of the IEEE International Conference on Computer Vision* (pp. 667–676). Venice, Italy: IEEE.. doi: https://doi.org/10.1109/ICCV.2017.79.

Lou, Y., Bai, Y., Lin, J., Wang, S., Chen, J., Chandrasekhar, V., ... & Gao, W. (2017, April). Compact deep invariant descriptors for video retrieval. In *2017 Data Compression Conference (DCC)* (pp. 420–429). Snowbird, UT: IEEE.. doi: https://doi.org/10.1109/DCC.2017.31.

Luo, D., Liu, C., Zhou, Y., Yang, D., Ma, C., Ye, Q., & Wang, W. (2020, April). Video cloze procedure for self-supervised spatio-temporal learning. *Proceedings of the AAAI Conference on Artificial Intelligence 34*(07), 11701–11708.

Markatopoulou, F., Galanopoulos, D., Mezaris, V., & Patras, I. (2017, June). Query and keyframe representations for ad-hoc video search. In *Proceedings of the 2017 ACM on International Conference on Multimedia Retrieval* (pp. 407–411). Bucharest, Romania: ACM..

Misra, I., Zitnick, C. L., & Hebert, M. (2016). Shuffle and learn: unsupervised learning using temporal order verification. In *European Conference on Computer Vision* (pp. 527–544). Cham. Amsterdam: Springer.doi: https://doi.org/10.1007/978-3-319-46448-0_32.

Muhammad, K., Obaidat, M. S., Hussain, T., Ser, J. D., Kumar, N., Tanveer, M., & Doctor, F. (2021). Fuzzy logic in surveillance big video data analysis: Comprehensive review, challenges, and research directions. *ACM Computing Surveys (CSUR)*, *54*(3), 1–33. doi: https://doi.org/10.1145/3444693.

Mühling, M., Korfhage, N., Müller, E., Otto, C., Springstein, M., Langelage, T., ... & Freisleben, B. (2017). Deep learning for content-based video retrieval in film and television production. *Multimedia Tools and Applications*, *76*(21), 22169–22194. doi: https://doi.org/10.1007/s11042-017-4962-9.

Mühling, M., Meister, M., Korfhage, N., Wehling, J., Hörth, A., Ewerth, R., & Freisleben, B. (2019). Content-based video retrieval in historical collections of the German broadcasting archive. *International Journal on Digital Libraries*, *20*(2), 167–183. doi: https://doi.org/10.1007/ s00799-018-0236-z.

Noroozi, M., & Favaro, P. (2016, October). Unsupervised learning of visual representations by solving jigsaw puzzles. In *European Conference on Computer Vision* (pp. 69–84). Cham, Amsterdam: Springer.doi: https://doi.org/10.1007/978-3-319-46466-4_5.

Pang, G., Shen, C., Cao, L., & Hengel, A. V. D. (2021). Deep learning for anomaly detection: A review. *ACM Computing Surveys (CSUR)*, *54*(2), 1–38. doi: https://doi.org/10.1145/3439950.

Paysan, D., Haug, L., Bajka, M., Oelhafen, M., & Buhmann, J. M. (2021). Self-supervised representation learning for surgical activity recognition. *International Journal of Computer Assisted Radiology and Surgery*, *16*(11), 2037–2044. doi: https://doi.org/10.1007/s11548-021-02493-z.

Podlesnaya, A., & Podlesnyy, S. (2016, September). Deep learning based semantic video indexing and retrieval. In *Proceedings of SAI Intelligent Systems Conference* (pp. 359–372). Cham : Springer.

Ram, R. S., Prakash, S. A., Balaanand, M., & Sivaparthipan, C. B. (2020). Colour and orientation of pixel based video retrieval using IHBM similarity measure. *Multimedia Tools and Applications*, *79*(15), 10199–10214. doi: https://doi.org/10.1007/s11042-019-07805-9.

Ren, S., He, K., Girshick, R., & Sun, J. (2016). Faster R-CNN: Towards real-time object detection with region proposal networks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *39*(6), 1137–1149.

Rui, Y., Huang, T. S., Ortega, M., & Mehrotra, S. (1998). Relevance feedback: A power tool for interactive content-based image retrieval. *IEEE Transactions on Circuits and Systems for Video Technology*, *8*(5), 644–655. doi: https://doi.org/10.1109/76.718510.

Rumelhart, D. E., Hinton, G. E., & Williams, R. J. (1985). Learning internal representations by error propagation. In: J. A. Anderson, & E. Rosenfeld. (eds.), *Neurocomputing: Foundations of Research* (pp. 673–695). MIT Press.

Shelhamer, E., Long, J., & Darrell, T. (2017). Fully convolutional networks for semantic segmentation. *IEEE Transactions on Pattern Analysis & Machine Intelligence*, *39*(04), 640–651.

Simonyan, K., & Zisserman, A. (2014). Two-stream convolutional networks for action recognition in videos. In *Proceedings of the 27th International Conference on Neural Information Processing Systems-Volume 1* (pp. 568–576). Montreal, Canada: MIT Press.

Soomro, K., Zamir, A. R., & Shah, M. (2012). UCF101: A dataset of 101 human actions classes from videos in the wild. *arXiv preprint arXiv:1212.0402*.

Subudhi, B. N., Rout, D. K., & Ghosh, A. (2019). Big data analytics for video surveillance. *Multimedia Tools and Applications*, *78*(18), 26129–26162. doi: https://doi.org/10.1007/s11042-019-07793-w.

Tran, D., Bourdev, L., Fergus, R., Torresani, L., & Paluri, M. (2015). Learning spatiotemporal features with 3D convolutional networks. In *Proceedings of the IEEE International Conference on Computer Vision* (pp. 4489–4497). Santiago, Chile: IEEE.. doi: https://doi.org/10.1109/ICCV.2015.510.

Tran, D., Wang, H., Torresani, L., Ray, J., LeCun, Y., & Paluri, M. (2018). A closer look at spatiotemporal convolutions for action recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (pp. 6450–6459). Salt Lake City, UT: IEEE..

Ueki, K., Hirakawa, K., Kikuchi, K., Ogawa, T., & Kobayashi, T. (2017, November). Waseda_Meisei at TRECVID 2017: Ad-hoc Video Search. In *TRECVID*.

Wang, J., Jiao, J., & Liu, Y. H. (2020, August). Self-supervised video representation learning by pace prediction. In *European Conference on Computer Vision* (pp. 504–521). Cham, Glasgow: Springer. .

Wang, L., Song, D., & Elyan, E. (2012, October). Improving bag-of-visual-words model with spatial-temporal correlation for video retrieval. In *Proceedings of the 21st ACM International Conference on Information and Knowledge Management* (pp. 1303–1312). Hawaii, USA: ACM..

Wu, J. Y., Tamhane, A., Kazanzides, P., & Unberath, M. (2021). Cross-modal self-supervised representation learning for gesture and skill recognition in robotic surgery. *International Journal of Computer Assisted Radiology and Surgery*, *16*(5), 779–787. https://doi.org/10.1007/s11548-021-02343-y.

Xu, D., Xiao, J., Zhao, Z., Shao, J., Xie, D., & Zhuang, Y. (2019). Self-supervised spatiotemporal learning via video clip order prediction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (pp. 10334–10343). Long Beach, CA: IEEE..

Yao, Z., Wang, Y., Long, M., Wang, J., Philip, S. Y., & Sun, J. (2020, July). Multi-task learning of generalizable representations for video action recognition. In *2020 IEEE International Conference on Multimedia and Expo (ICME)* (pp. 1–6). London, UK: IEEE..

Zagoruyko, S., & Komodakis, N. (2016). Paying more attention to attention: Improving the performance of convolutional neural networks via attention transfer. In *International Conference on Learning Representations (ICLR)*.

Zhou, W., Li, H., & Tian, Q. (2017). Recent advance in content-based image retrieval: A literature survey. *arXiv preprint arXiv:1706.06064*.

Zhu, Y., Huang, X., Huang, Q., & Tian, Q. (2016). Large-scale video copy retrieval with temporal-concentration sift. *Neurocomputing*, *187*, 83–91. doi: https://doi.org/10.1016/j.neucom.2015.09.114.