LITERATURE REVIEW OF RESEARCH PAPERS ON VIDEO CLASSIFICATION

Paper 1: Dance Gesture Recognition: A Survey

By: Mampi Devi, Sarat Saharia, and D.K. Bhattacharya

International Journal of Computer Applications (0975 – 8887)
Volume 122 – No.5, July 2015

Gesture recognition means the identification of different expressions of human body parts to express the idea, thoughts and emotion. It is a multi-disciplinary research area. The application areas of gesture recognition have been spreading very rapidly in our real-life activities including dance gesture recognition. Dance gesture recognition means the recognition of meaningful expression from the different dance poses. Today, research on dance gesture recognition receives more and more attention throughout the world. The automated recognition of dance gestures has many applications. The motive behind this survey is to present a comprehensive survey on automated dance gesture recognition with emphasis on static hand gesture recognition. Instead of whole body movement, considered human hands because human hands are the most flexible part of the body and can transfer the most meaning. A list of research issues and open challenges is also highlighted.

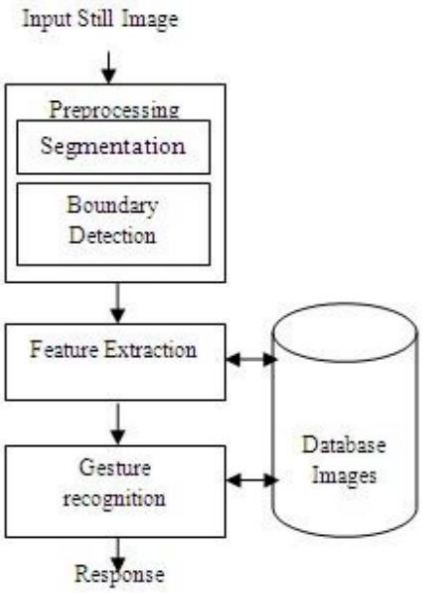
In this paper, firstly the dance gestures are recognized. Firstly, images are recorded and processed by resizing, cropping and filtering. Then on the objects are recognized from the background. Then the boundaries of each object is detected, by erosion, dilation, thinning, filtering and filling operations. Object detection is an important feature, which needs to be considered then after. Finally, features are extracted from these images. The final goal is to find out an optimal feature. From there on, by Supervised and unsupervised learning, using Artificial Neural Network, Support Vector Machine, KNN, back propagation neural network etc. Gestures are classified by different methods such as Kendon Classification, Efron's Classification, David McNeil Classification, and David Karam's Classification etc.

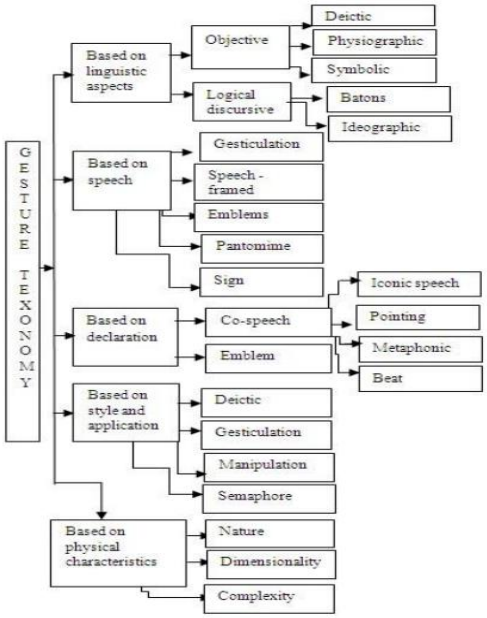
A further literature review is done in this paper. It is suggested that video classifications are still in its first stages, while it is gathering momentum. This paper also comes along with gesture analysis of the hands. Recognizing them along with the moves is tough. Then the paper provides distinguished features upon which the dance categories can be recognized. The algorithm is based on an understanding of human body instruction. It represents all the 22 physical segmentation of human anatomy and each of which has independent movement. Here, all parent segments inherit the combined characteristics of all child segments. The algorithm considered the variation of gestures from one choreographer to others. However, this algorithm is limited to a finite set of pose states. Another popular algorithm found in this domain based on torso frame for human joint is known as Skeletal Tracking Algorithm (STA). The algorithm represents all the angular skeleton and mapping the skeleton motion data into a smaller set of features. The torso frame is computed as a basis of co-ordinate of the other skeleton joint.

The work based on important joint features like left/right foot and left /right elbow of Bali traditional dance. The authors of jointly propose gesture recognition algorithm for Indian classical dance style using sensor. The authors made one device, which generates the skeleton of human body from which twenty different junction 3-dimensional coordinates are obtained. The authors use a unique system and extract the features to distinguish anger, fear, happiness, sadness and relaxation. They calculate the distance between different parts of the upper human body and generate velocity, acceleration along with the angle between different angles. Based on that they extract twenty-three features. The performance of their method is almost 86.8%.

Tree structured Bayesian network and expectation maximization (EM) algorithm with K-means clustering was applied to calculate and classifying the poses. Finally, Hidden Markov Model (HMM) is used for recognition as per the literature review; the dance gesture recognition approach has been divided into two approaches:

1. Glove based Approach: In this approach, sensor devices and hand gloves are used in the image acquisition phase. It provides the co-ordinate points of skeleton and orientation
2. Vision Based Approach: In this approach, the images are captured by camera. It is a very simple approach deals with the simple image characteristics like colour, texture and intensity values.





Existing methods such as Statistical Methods, Finite State Machine, Soft computing methods, Hybrid method, Back-propagation neural network etc. have been used to handle the gesture recognition problem.

Dances express human love, emotions, devotion, narrate stories of religious scriptures, and are integral parts of the celebration of life. In this survey paper, the overview of gestures recognition with special emphasis on hand and dance gestures are discussed. This paper also provides a comprehensive study on vision based and glove based gesture recognition, gesture taxonomy. Finally, this survey paper concludes with existing approaches and methods to implement and scope for future work.

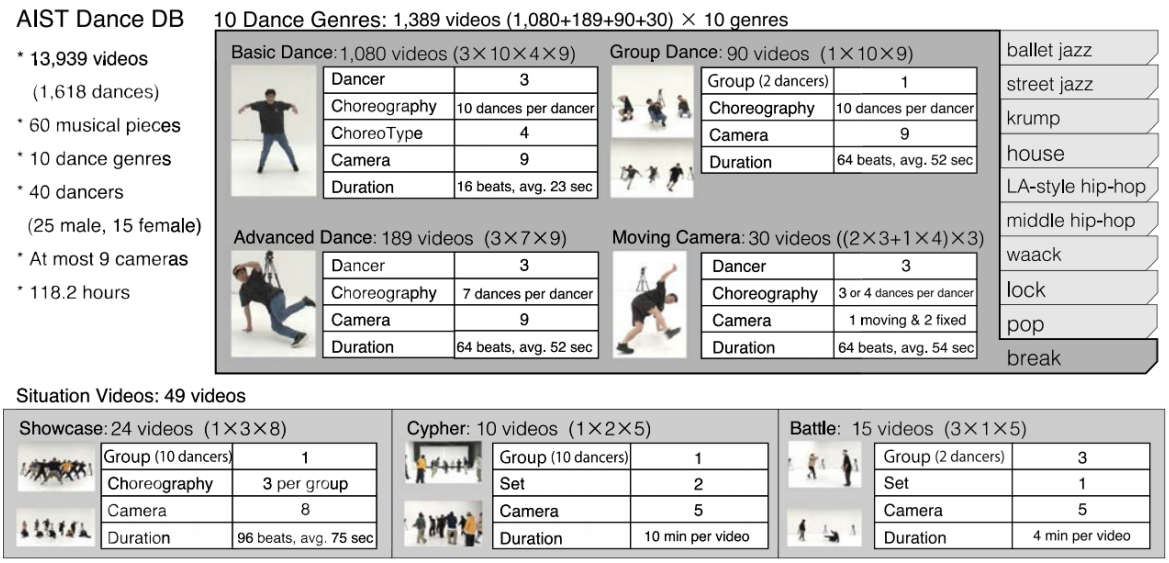
Paper 2: AIST Dance Video Database: Multi-Genre, Multi-Dancer, And multi camera database for Dance Information Processing

By: Shuhei Tsuchida, Satoru Fukayama, Masahiro Hamasaki, and Masataka Goto

Proceedings of the 20th ISMIR Conference, Delft, Netherlands
November 4-8, 2019

The AIST Dance DB as the first large-scale shared database focusing on street dances to facilitate research on a variety of tasks related to dancing to music. It consists of 13,939 dance videos covering 10 major dance genres as well as 60 pieces of dance music composed for those genres. The videos were recorded by having 40 professional dancers (25 male and 15 female) dance to those pieces. Carefully designed, this database so that it can cover both solo dancing and group dancing as well as both basic choreography moves and advanced moves originally choreographed by each dancer. Moreover, multiple cameras are used surrounding a dancer to simultaneously shoot from various directions. The AIST Dance DB will foster new MIR tasks such as dance-motion genre classification, dancer identification, and dance-technique estimation. A dance-motion genre-classification task is performed and developed four baseline methods of identifying dance genres of videos in this database. These methods are evaluated by extracting dancer body motions and training their classifiers based on long short-term memory (LSTM) recurrent neural network models and support-vector machine (SVM) models.

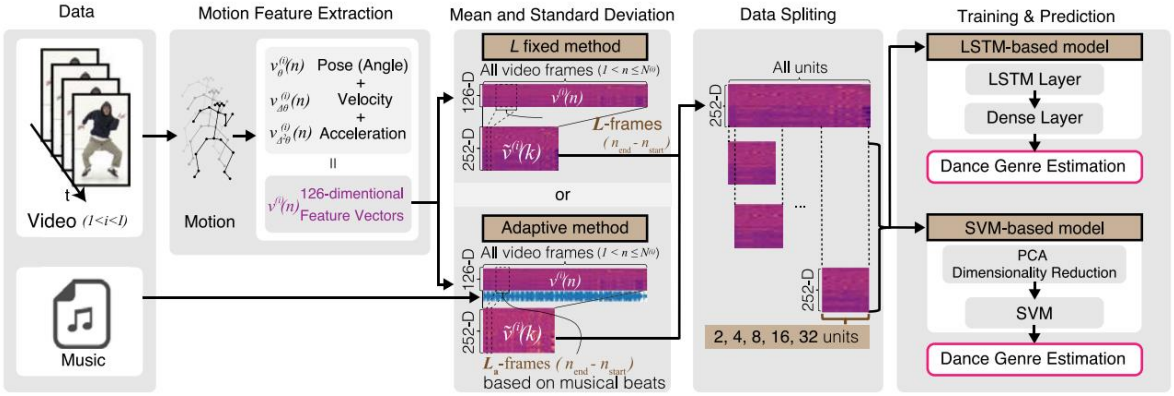
The dance videos are captured by various shooting angles and of various dance genres. The dataset consists of 210 dance videos shot from the front camera only and covers the 10 dance genres. Each genre has 21 dance videos by three dancers, each of whom uses seven original choreographies. In total, the dataset covers 210 different choreographies by 30 different dancers.



Each method is trained to classify an excerpt of the input video into one of the 10 dance genres. In the first step of motion-feature extraction, we use the OpenPose library to estimate the dancer's skeleton (body pose and motion) in all video frames (60 frames per second). This can reduce the dependency on the AIST Dance DB since the estimated body pose and motion do not have original RGB pixel information. Each joint angle is then converted into two dimensional values θx and θy by calculating the sine and cosine of the angles to make the distance calculation between angular values easier. As a result, we convert the 21-dimensional angular values into a 42-dimensional feature vector.

$$v^{(i)}_{\Delta\theta}(n) = v^{(i)}_{\theta}(n) - v^{(i)}_{\theta}(n-1),$$

$$v^{(i)}_{\Delta^2\theta}(n) = v^{(i)}_{\Delta\theta}(n) - v^{(i)}_{\Delta\theta}(n-1).$$

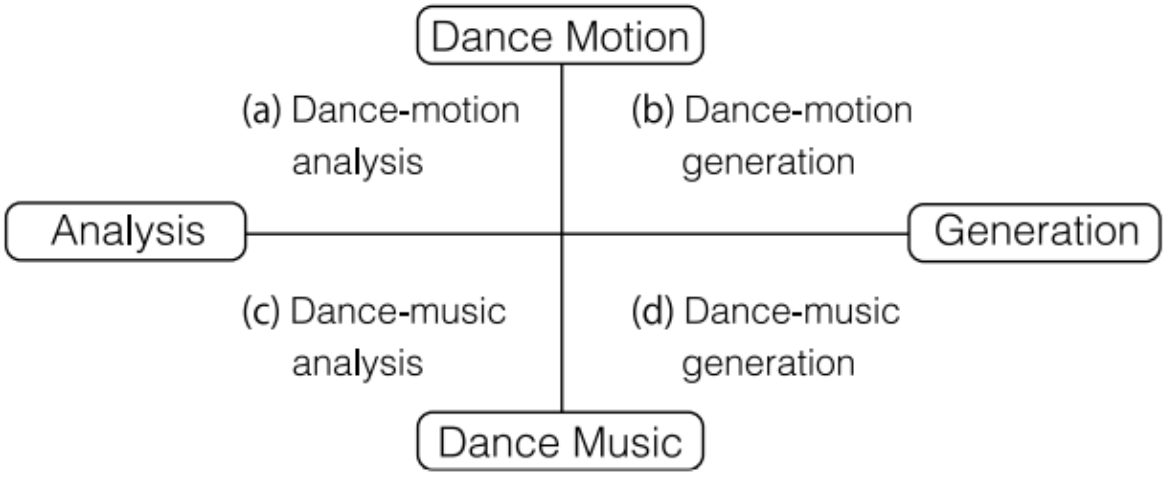


For the LSTM-based model, a bi-directional recurrent neural network (RNN) is used with one layer of LSTM cell. The network outputs a 10-dimensional one-hot vector representing dance genres. A rectified linear unit activation function is applied to the output of the LSTM. Batch normalization is applied to the output layer of the dense layers. Cross entropy is taken as the loss function and a batch size of 10. Then training this model with a learning rate of 5e−4 through 100 epochs and record the trained model at the minimum validation loss.

For the SVM-based model, firstly 200-dimensional vectors is obtained by using principal component analysis to reduce the dimension of the training data, then train the SVM model. Finally, the dance genre of every window-level feature vector in a video by using these two models is estimated.

The best genre-classification accuracy was 91.4% when the L-fixed method with the LSTM-based model where the number of frames was 60 and the number of units was 32 was used. In the case of the L-fixed method with

the SVM-based model, however, the best accuracy was dropped to 84.0%. The dance-motion genre classification can be executed with an accuracy of 56.6% by using only 0.67 sec corresponding to 40 frames (20 frames per unit × 2 units) of a video. This was much shorter than we expected. . With a small number of frames and units, the classifier easily confused street jazz and ballet jazz and the estimation accuracy of house dropped. This can be understood from the fact that street jazz and ballet jazz contain similar poses and house contains many movements that are commonly found in other dance genres, such as simple lateral movements.



The main contributions of this work are threefold:

1) Built the first large-scale shared database containing original street dance videos with copyright-cleared dance music.
2) Proposed and discussed a new research area dance information processing.
3) Proposed a dance-motion genre-classification task and developed four baseline methods. The AIST Dance DB will help researchers develop various types of dance-information-processing technologies to give academic, cultural, and social impact.

Paper 3: Indian Classical Dance Action Identification and Classification with Convolutional Neural Networks

By: P.V.V. Kishore, K.V.V. Kumar, E. Kiran Kumar, A.S.C.S. Sastry, M. Teja Kiran, D. Anil Kumar, and M. V. D. Prasad

Hindawi, Advances in Multimedia Volume 2018, Article ID 5141402, 10 pages
Published 22 January 2018

Automatic human action recognition is a complicated problem for computer vision scientists, which involves mining and categorizing spatial patterns of human poses in videos. Human action is defined as a temporal variation of human body. The last decade has seen a jump in online video creation and the need for algorithms that can search within the video sequence for a specific human pose or object of interest. The problem is to extract and identify a human pose and classify it into labels based on trained CNN feature maps. The objective of this work is to extract the feature maps of Indian classical dance poses from both online and offline data. The created offline dataset is having 200 Indian classical dance mudras/poses performed by 10 native classical dancers (i.e., 10 sets) at a rate of 30 frames per second (fps). Training is initiated with three different batch sizes. In Batch I of training there is only one set, that is, 200 poses performed by one dancer for 2 seconds each at 30 fps, total of 200 × 1 × 2 × 30 = 12000 dance pose images. Batch II of training is done using five sets, that is, a total of 200 × 5 × 2 × 30 = 60000 dance pose images. In Batch III of training, eight sets of sign images were used. The trained CNNs are tested with two discrete video sets having different dance performers with varying backgrounds. The robustness testing is performed in two cases. In Case I of testing, the same dataset, that is, an already trained dataset, is used and in Case II of testing different dataset is used. The similar training and testing are done on online data also. Figure 1 shows the sample database created for this work. The performance of the

CNN algorithms is measured for both online and offline data, based on their accuracy in recall and recognition rates. A part of the database is shown as example:

In this paper, Indian classical dance is discussed. Indian classical dance forms are practiced for 5000 years worldwide. However, it is difficult for a dance lover to fully hold the content of the performance as it is made up of hand poses, body poses, and leg movements, hands with respect to face and torso, and finally facial expressions. All these movements should synchronize in precision with both vocal song and the corresponding music for various instruments. Apart from these complications, the dancer wears complicated dresses with a nice makeup and at times during performance; the backgrounds are changing depending on the story, which truly makes this an open-ended problem. Firstly, ANN and SVMs were used in order to distinguish features between images. However, they were not of much use, as they were incompetent. Then CNNs came into use. From there on, CNNs are majorly used. Deep CNN is well suited for this kind of Video, Image Recognition.

In this paper, a novel CNN based Indian classical dance identification method is proposed to achieve higher recognition rates. Different CNN architectures are implemented and tested on our dance data to bring out the best architecture for recognition. Tree different pooling techniques, namely, mean pooling, max pooling, and stochastic pooling, are implemented and stochastic pooling was found to be the best for our case. To prove the capability of CNN in recognition, the results are compared with the other traditional state-of-the-art techniques SVM, AGM, ANN, and deep ANN. The architecture used is:

Layer (type)	Function	Output shape
Input		3 × 640 × 480
conv1	Convolution	16 × 128 × 128
activation_1	Activation	16 × 128 × 128
conv2	Convolution	16 × 120 × 120
activation_2	Activation	16 × 120 × 120
stoch_pooling_1	Stochastic pooling	16 × 60 × 60
dropout_1	Dropout	16 × 60 × 60
conv3	Convolution	32 × 56 × 56
activation_3	Activation	32 × 56 × 56
conv4	Convolution	32 × 52 × 52
activation_4	Activation	32 × 52 × 52
stoch_pooling_2	Stochastic pooling	32 × 26 × 26
dropout_2	Dropout	32 × 26 × 26
flatten_1	Flatten	4608 × 1 × 1
dense_1	Fully connected	32 × 1 × 1
activation_5	Activation	32 × 1 × 1
dropout_3	Dropout	32 × 1 × 1
dense_2	Fully connected	28 × 1 × 1
activation_6	Activation	28 × 1 × 1
Output	SoftMax regression	28 × 1 × 1

The architecture design looks like:

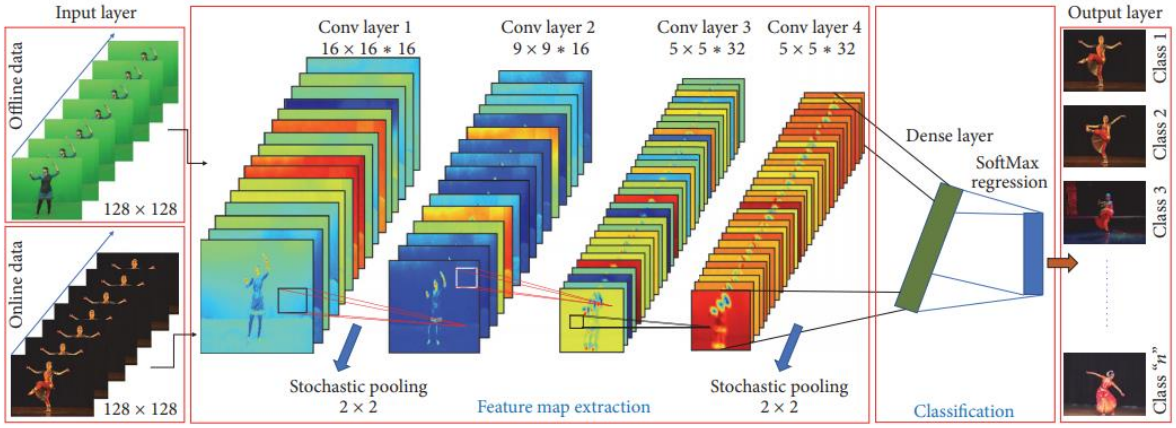
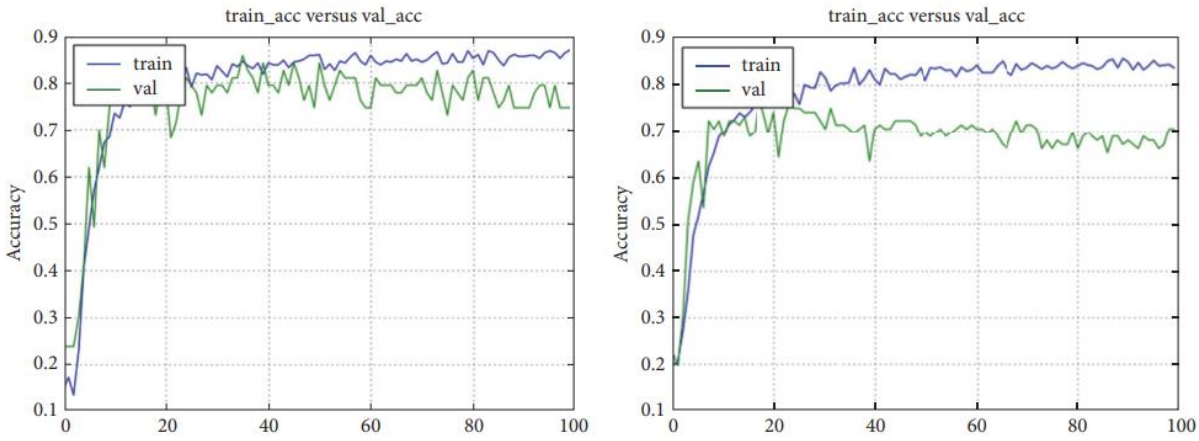


FIGURE 2: The proposed Deep CNN architecture.

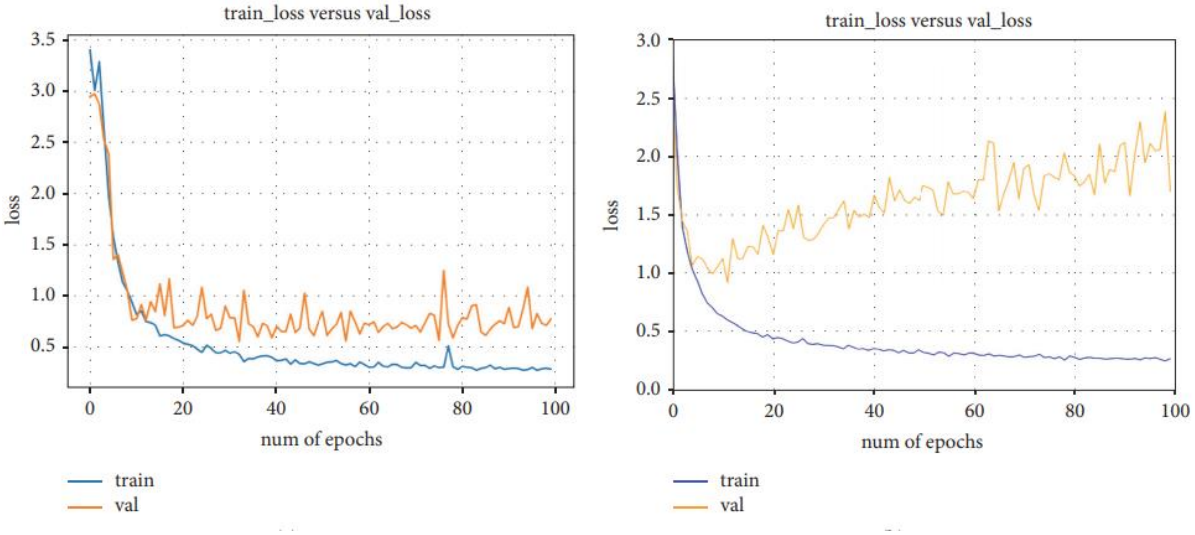
The network is trained to learn the features of each dance pose by means of a supervised learning. The internal feature representation reflects the likeness among training samples. We outline 200 poses from ICD (offline data) performed by 10 classical dancers. The size of the total dataset is 2000 dance poses with each pose recording normalized to 2 secs or 60 frames per second forming 120k frames. Similarly, the online dance data is downloaded from YouTube and each pose normalized to 60 fps. All together to know the feature representation learned by the CNN system, the maximized activation neuron is extracted to recognize the dance pose accurately. Finally, the feature maps were visualized by averaging the image patches with stochastic response in higher layers. The results for different datasets used:

Training batch	Number of training datasets	Training datasets	Testing datasets	Recognition rates (%)	
				Offline data	Online data
Batch I (12000 images)	1	Dataset 1	Dataset 1	92.12	90.34
			Dataset 2	90.88	88.98
Batch II (60000 images)	5	Dataset 1 to Dataset 5	Dataset 1	92.83	91.52
			Dataset 6	91.17	89.87
Batch III (96000 images)	8	Dataset 1 to Dataset 8	Dataset 1	93.33	91.96
			Dataset 9	91.47	89.92
			Dataset 10	91.99	89.05

The training and validation accuracies for Online and Offline datasets:



The training and validation losses for Online and Offline datasets are as follows:



Further, other forms of neural networks are compare with their accuracies on the Online and Offline datasets:

Table 4: Recognition rates of offline dance data identification with different classifiers.

Classifier	Average recognition rates (%) of offline dance data					
	Batch I training		Batch II training		Batch III training	
	Testing with the same dataset	Testing with different dataset	Testing with the same dataset	Testing with different dataset	Testing with the same dataset	Testing with different dataset
ANN [15]	83.42	80.01	85.03	81.61	86.49	82.11
Deep ANN [16]	87.39	83.62	88.01	84.92	89.49	86.99
SVM [13]	75.93	71.31	78.35	74.48	80.46	76.78
Adaboost [14]	80.19	76.49	80.98	77.29	81.76	79.09
AGM [17]	87.20	83.16	87.89	84.63	88.23	85.05
LeNet [25]	88.14	85.49	88.55	86.32	87.92	86.85
VGG [27]	89.98	87.02	90.12	88.41	88.76	88.02
Our proposed CNN architecture	92.14	90.88	92.83	91.17	93.33	92.47

Table 5: Recognition rates of online dance data identification with different classifiers.

Classifier	Average recognition rates (%) of online dance data					
	Batch I training		Batch II training		Batch III training	
	Testing with the same dataset	Testing with different dataset	Testing with the same dataset	Testing with different dataset	Testing with the same dataset	Testing with different dataset
ANN [15]	82.82	79.13	84.16	80.46	85.95	81.19
Deep ANN [16]	85.27	81.26	86.45	82.91	87.43	84.87
SVM [13]	74.12	70.66	77.63	73.45	79.51	74.23
Adaboost [14]	80.01	74.54	79.49	76.62	80.55	78.10
AGM [17]	85.11	80.09	86.09	81.31	86.88	88.41
LeNet [25]	86.43	81.79	88.26	80.49	87.73	85.66
VGG [27]	88.52	84.54	89.15	85.45	88.19	86.76
Our proposed CNN architecture	90.34	88.98	91.52	89.87	91.96	89.92

CNN is a powerful artificial intelligence tool in pattern classification. In this paper, we proposed a CNN architecture for classifying Indian classical dance poses/mudras. The CNN architecture is designed with four convolutional layers. Each convolutional layer with different filtering window sizes is considered which improves the speed and accuracy in recognition. A stochastic pooling technique is implemented which combines the advantages of both max and mean pooling techniques. Training accuracy and validation accuracies for this CNN architecture are better than the previously proposed ICD classification models. A less amount of training and validation loss is observed with the proposed CNN architecture.

Paper 4: Automatic Video Classification: A Survey of the Literature

By: Darin Brezeale and Diane J. Cook

IEEE TRANSACTIONS ON SYSTEMS, MAN, AND CYBERNETICS—PART C: APPLICATIONS AND REVIEWS, VOL. 38
3, MAY 2008

That automated methods of classifying video are an important and active area of research is demonstrated by the existence of the TRECVid video retrieval benchmarking evaluation campaign, which began as a track of the Text Retrieval Conference (TREC) before branching off on its own. TRECVid provides data sets and common tasks that allow researchers to compare their methodologies under similar conditions. While much of the TRECVid is devoted to video information retrieval, video classification tasks exist as well, such as identifying clips containing faces or on-screen text, distinguishing between clips representing outdoor or indoor scenes, or identifying clips with speech or instrumental sound. A large number of approaches have been attempted for performing automatic classification of video. After reviewing the literature of methods, we found that these approaches could be divided into four groups: text-based approaches, audio-based approaches, visual-based approaches, and those that used some combination of text, audio, and visual features. Most authors incorporated a variety of features into their approach, in some cases from more than one modality. Therefore, in addition to describing the general features utilized, we also provide summaries of the papers we found published in this area.

In a review of the video classification literature, we found many of the standard classifiers, such as Bayesian, support vector machines (SVM), and neural networks. However, two methods for classification are particularly popular: Gaussian mixture models (GMMs) and hidden Markov models (HMMs). Because of the iniquitousness of these two approaches, we provide some background on the methods here. Researchers who wish to use a probabilistic approach for modelling a distribution often choose to use the much-studied Gaussian distribution. A Gaussian distribution, however, does not always model data well. One solution to this problem is to use a linear combination of Gaussian distributions, known as a GMM.

Firstly, audio based features are considered, where Zero crossing rate, frequency based domain has been used to calculate the audio features. MFCCs are calculated and used in order to carry out the feature processing of the audio data. As the dance is based on music, so by classifying audio genres in the videos, the videos can be classified. Discrete cosine transformations are used in order to get the features. Up next Visual based features are dealt with. Here, we consider the optical flow, object detection and colour based feature detection. Also features are detected on the basis of motion and angle of shot of the specific frame in the video. There are a number of papers that deal with these kinds of approaches and they have been ranked as the following:

	Color-Based			Shot-Based		Object-Based			MPEG	Motion-Based			
Paper	Color	Texture	Edge	Trans.	Length	Face	Text	Other	DCT	Motion Vectors	Optical Flow	Frame Diffs	Other
Iyengar and Lippman [59]					X						X	X	
Girgensohn and Foote [74]	X												
Wei et al. [56]				X		X	X						
Dimitrova et al. [66]						X	X						
Truong et al. [60]	X			X	X						X		
Kobla et al. [7]							X			X			
Roach et al. [75]												X	
Roach et al. [76]											X	X	
Lu et al. [64]	X												
Jadon et al. [63]				X	X						X		
Hauptmann et al. [2]	X	X	X										
Pan and Faloutsos [39]	X												
Rasheed et al. [62]	X				X								
Gibert et al. [77]	X									X			
Yuan et al. [65]	X				X	X							X
Hong et al. [78]	X	X									X		
Brezeale and Cook [18]									X				
Fischer et al. [35]	X			X	X						X	X	
Nam et al. [4]	X						X						X
Huang et al. [36]	X									X			
Qi et al. [21]	X												
Jasinschi and Louie [19]	X		X			X	X		X				
Roach et al. [42]												X	
Rasheed and Shah [40]	X				X								X
Lin and Hauptmann [20]	X												
Lee et al. [37]			X						X				
Wang et al. [13]					X	X	X			X			
Xu and Li [43]	X	X								X			
Fan et al. [8]	X	X						X					

There onwards, only the video features are taken into consideration and they are trained by HMM. The classes were taken as News and sports, and frame-wise tracking was done to these videos and using the network the classification was done. Also afterwards, text based classification was done on these types of videos. GMM and SVM based approaches were also tested and score. A comparison of features table should make things cear.

Feature Type	Pros/Cons
Text Features	
Closed Captions	High accuracy when not produced in real-time, high dimensionality, computationally cheap to extract
Speech Recognition	High error rates
OCR	Can extract video text not present in dialog, computationally expensive
Audio Features	Require fewer computational resources than visual features, clips are typically shorter in length and smaller in file size than video clips, difficult to distinguish between multiple sounds
Visual Features	
Color-Based	Simple to implement and process, crude representation
MPEG	Easy to extract, but video must be in MPEG format
Shot-Based	Difficult to identify shots automatically, so may not be accurate
Object-Based	Difficult, limited on number of objects, computationally expensive
Motion-Based	Difficult to distinguish between types of motion, computational requirements range from low (MPEG motion vectors, frame differencing) to high (optical flow)

Features are drawn from three modalities: text, audio, and visual. The majority of the literature describes approaches that utilize features from a single modality. While much has been done, there are still many research opportunities in automatic video classification and the related field of video indexing. Only a few of the papers reviewed attempted to perform classification at the shot or scene level. Being able to classify at the shot or scene level has many applications, such as content filtering (e.g., identifying violent scenes), identification of important scenes, and video summarization. This would also be useful in subdividing genre, such as creating a category of action movies that include car chases.

Paper 5: A review of Machine Learning techniques used for Video classification

By: Seetha Parameswaran, Dr. Shelbi Joseph

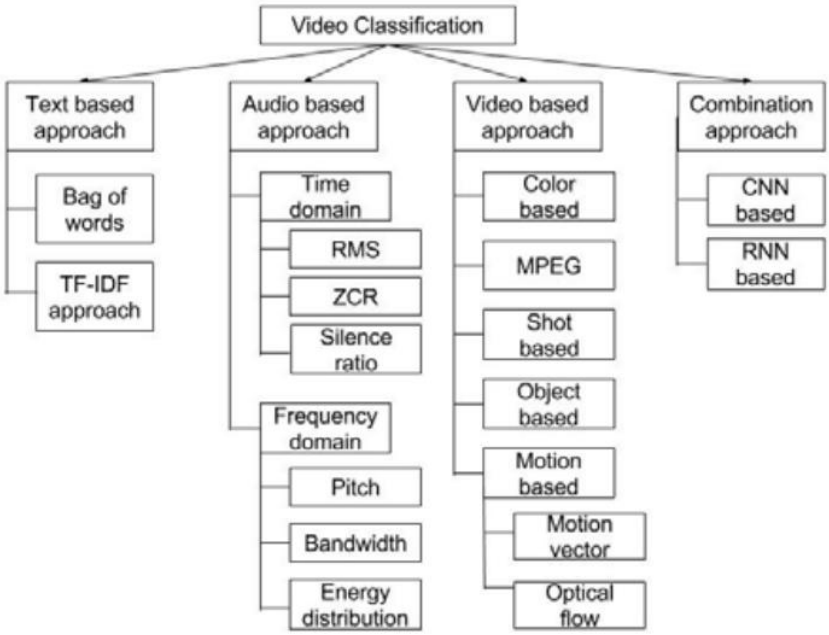
INTERNATIONAL JOURNAL OF CURRENT ENGINEERING AND SCIENTIFIC RESEARCH (IJCESR)
ISSN (PRINT): 2393-8374, (ONLINE): 2394-0697, VOLUME-4, ISSUE-12, 2017

With the widespread penetration of Internet worldwide and with the improved availability of cheap connectivity and storage, users have shifted towards using more and more video in their daily use. The large penetration of Social media like YouTube, Facebook, Instagram, etcetera and decentralized communication paradigms like WhatsApp has significantly altered user behaviour, with multimedia content being generated and shared with greater ease than in the past. Video provides a much richer experience to the user and enables users to visualize progression of events over time. Thus, it is easier to show using video how to prepare a chocolate cake instead of giving a written text of its recipe. Alternatively, it is easier to understand a lecture from video than it is to read from text. Video contains both spatial and temporal information. Objects in a video carry meaning by the way they are ordered in a frame. They also carry additional meaning when they move or do not move from one frame to another. While extracting information from a video, it is imperative to identify and classify objects and their transformation. The feature vector and the classifier determine the effectiveness of a video classification system. This paper attempts to review the various feature extraction methods used for video classification and the classifiers used. Many papers in the area of video classification published in the past years were reviewed. This work will be helpful to those who intend to start a research in video classification to have an insight into the popular existing works on the field. The contribution of this paper is organized in the following sections as follows; the section II of this paper discusses the various feature extractors and the classifiers used. The next section lists the taxonomy. Finally, in section IV the conclusion and future scope is presented.

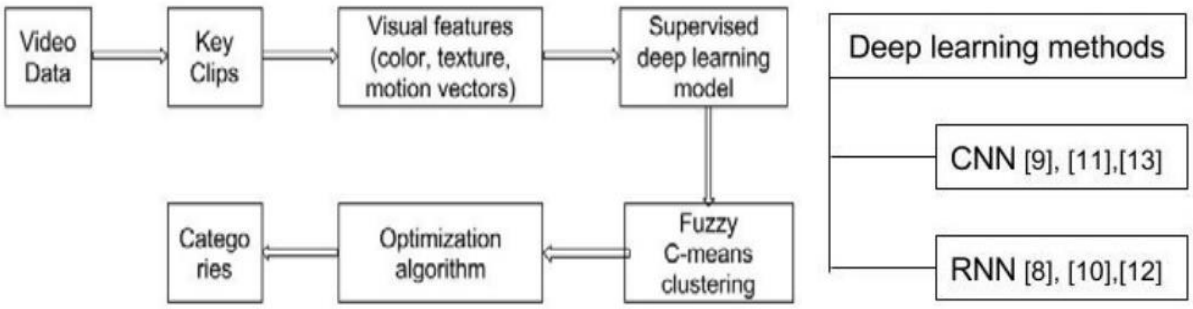
In this paper, the history of video classification procedure is discussed. Starting from Hidden Markov Models, the classifications were started. With discovery of new features, the HMM works better. Only some specific features are used and classification is performed. Only then, the classification is done properly. Then GMM is used to classify video clips. The video clips are used to extract features using foreground and background object motion detection. Similarly, SVM is used and classified. Spatio-temporal audio-visual feature vector and processes it using Principal Component Analysis. Using Expectation-Maximisation (EM) algorithm, the

parameters of GMM are estimated. A Recurrent Neural Network (RNN) containing Long Short-Term Memory (LSTM) neurons is trained using SIFT features to categorize actions in sports video sequences. With the advancement in deep learning models and architecture, the feature extraction is automated. 3D Convolutional Neural Networks were used action recognition. 3D convolutions effectively incorporate spatial and motion information. Long-term RNN models directly map variable-length video frames to variable length natural language text and model complex temporal dynamics. Two-stream CNN architecture, 1 for spatial features and other for temporal features, is used. The work uses rank pooling coupled with CNN on activity and action recognition tasks. The rank pooling encodes temporal information by ordering frames of videos in a chronological order of their appearance.

This is the taxonomy of Video classification algorithms:



Four types of methods can do classification of videos. Text based approach considers OCR detection of characters in text. This initiates good feature learning. However, the disadvantage is that the text cannot encode the videos correctly. Closed captions are not available for all videos. Generation of transcript is difficult in dialog less videos. Up next, audio-based classification is talked about. In this approach, MFCCs, ZCR and Silence Ratios are used as features in learning process. Audio feature recognition are computationally less expensive, but still audio features cannot exactly encode the video data. Visual based approach relies on visual elements for video classification, either alone or along with text and audio features. Some combinations of text, audio and visual features are used. A feature super vector is build using convolution or product methods. Most commonly used architecture for a combination approach is by using deep learning method.



The methods reviewed so far in this paper uses unsupervised learning to develop video classification methods. This work uses the strength of unsupervised learning method and fuzziness to attain better classifications. In

this proposed work the video data is first divided into key clips and then we extract the visual features like colour, texture and motion vectors. These features are then subjected to supervised deep learning models and the then results then undergo Fuzzy C means clustering. With the objective of further improving the results we apply an optimization algorithm which then categorizes the video under a label.

Paper 6: A review of Video classification techniques

By: Mittal C. Darji, Dipti Mathpal

International Research Journal of Engineering and Technology (IRJET)
Volume: 04 Issue: 06 | June -2017

Video classification literature has been reviewed and techniques for the same are provided here in this paper. Classification process in general requires features based on which one can distinguish among the categories. These features are mainly taken from text, audio or visual content of the video. Based on that mainly three classification techniques are there as discussed here. Based on the application user has to select the method and features. Pros and cons of each method are mentioned in this paper with suitable applications. Video classification has been used to classify videos into categories like sports, comedy, news, dance, horror etc. Some researchers have also classified a single video into parts of different categories. All these classifications require the characteristics, which differ for each category. These characteristics are called features.

Classification Method	Feature Type	Advantages / Disadvantages	Application
Text Based Classification	OCR Closed Captions Speech Recognition	Computationally expensive Higher dimensionality Higher error rate	Reading score board Providing subtitles Reading headlines form news video
Audio Based Classification	Physical Features Perceptual Features	Shorter in length and size Computationally cheaper Difficult to differentiate similar sounds	Classifying movie into dialogs and songs Classifying videos into horror, action, comedy Classifying video into speech, music, environmental sound
Video Based Classification	Color-Based Features Shot-Based Features Object-Based Features	Larger size Computationally expensive Preprocessing is required Difficult to identify shots, not accurate	Object tracking Video summarization Separating news video and sight scenes Classification of different sports videos

Here also, three types of classification models are primarily focussed on. The first one is text based. Here the video can contain text on it, or it can be extracted from the speech. However, the encoding is not exact. Thus, the learning is not sufficient. OCR based character recognition is used.

Audio-based classification is the second method. Sample rate, overlap length and window length is specified. Amplitude values of sampled signal are directly used to compute these feature values. Such features are Zero Crossing Rate (ZCR), Short Time Energy (STE), Spectral Roll-off, Spectrum Centroid, Spectral Flux, Fundamental Frequency, Mel-Frequency Cepstral Coefficient (MFCC) etc. Psychological acoustic model is proposed that measures the perceptual features of sound based on the human perceptual system for sound. Perceptual

features such as Loudness, Pitch and Timbre are used, out of them Loudness and pitch are mostly considered. Music is having higher continuous amplitude than speech. Music also contains higher ZCR than speech due to a frequent variation in amplitude.

Visual features are mostly extracted from the frames of video or from the shots of video. Basic construction of a video is like: fundamental part of video is a frame. Hence, video can be called as a collection of frames. There are three methods of classifying, such as Colour based feature, shot-based feature, object-based feature. Colour based detections of features in images are used for learning. Nevertheless, colours cannot detect exact features for training, so this is not the best method for video classification. However, for images shot from different angles, positions, shot-based features can be extracted. This helps identifying the shot types. Finally, for a continuous video with moving frame, the objects in motion are detected and based on that learning is done.

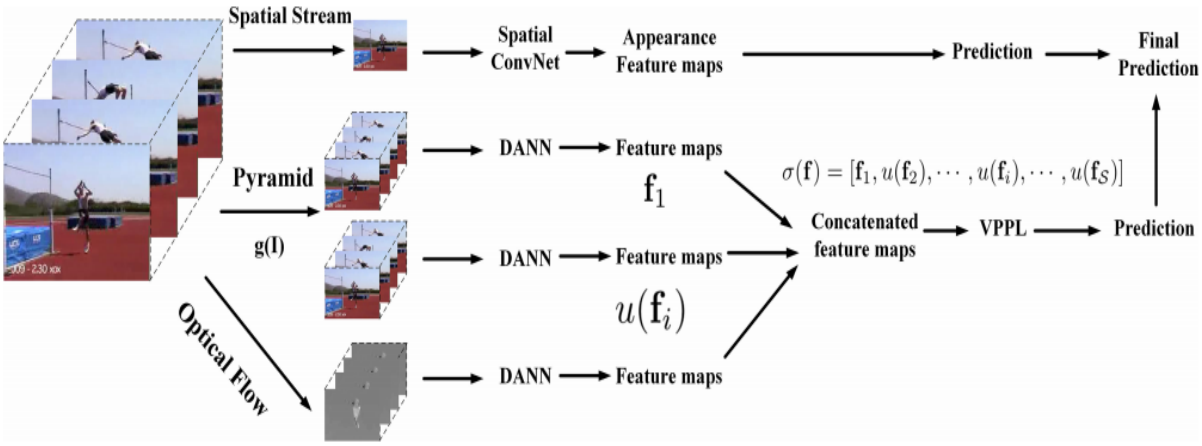
Paper 7: Multiscale Deep Alternative Neural Network for Large-Scale Video Classification

By: Jinzhou Wang, Wenmin Wang

IEEE TRANSACTIONS ON MULTIMEDIA, VOL. 20, NO. 10, OCTOBER 2018

In this paper, we first propose a novel deep alternative neural network (DANN) to mine rich context hierarchies for video classification. The DANN stacks alternative layers (ALs) consisting of a volumetric convolutional layer and a recurrent layer. The alternative deployment is used to preserve the contexts of local features in each layer and embed their evolutions in the hierarchical feature learning procedure. We demonstrate the advantages over standard feed-forward architectures in terms of context mining. In addition, we develop a much deeper version of the DANN by introducing a vertical dropout with skip connection to stack more ALs. We explain its benefits in terms of context exploration, faster convergence and reduction in training time. Additionally, we develop a multi-scale manner to construct DANN at various scales to further exploit complex context hierarchies, which can provide a remarkable improvement. We share good practices for applying multi-scale DANN to video classification, including input-output configurations and training testing methods. To verify the effectiveness of our method on general video classification, we contribute a new natural disaster video dataset and make it publicly available. The experimental results obtained on four large-scale datasets demonstrate the effectiveness of our method for both human activities and natural events.

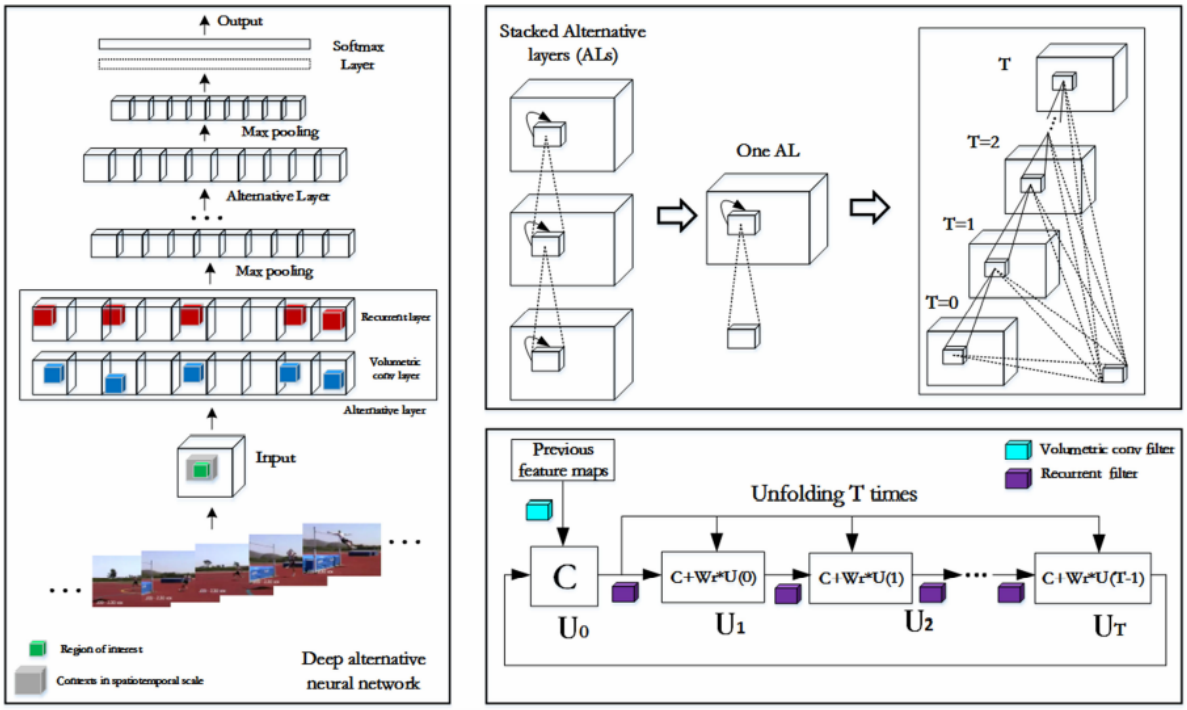
The standard approach to video classification with handcrafted solutions involves three major stages: First, local visual features that describe a region of a video are extracted either densely or sparsely. Next, the features are combined into a fixed-sized video-level description using quantization techniques. Finally, a classifier is trained on video level representation to distinguish among the visual classes of interest. In particular, first extended the 2D Harris corner detector to obtain representative tubes in 3D space. Since then, many 2D local descriptors have been extended to 3D to achieve video understanding, such as 3D SURF, HOG3D and 3D SIFT.

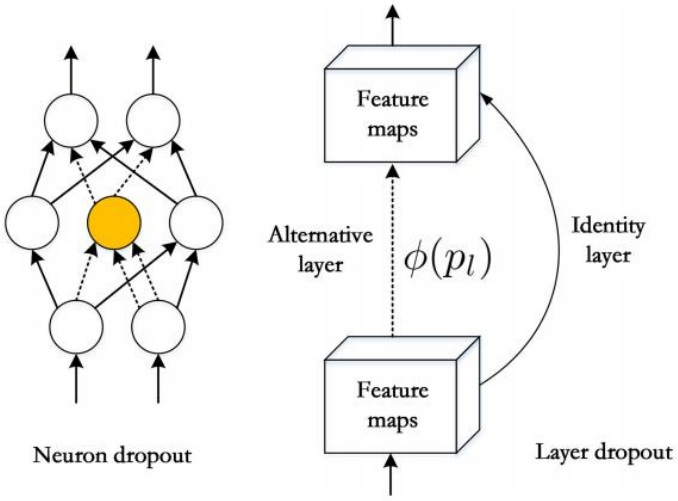


For deep learning CNN and RCNN are used. The images are used to identify features firstly. Then those features are used in order to make the dataset for training the network. ImageNet is used in order to train a deeper network for a better learning. Recurrent layers are provided in between alternate CNN layers and therefore building the RNN is used to train and classify the videos.

Similar to popular deep neural networks, our DANN can be improved by going deeper. Many studies have been devoted to architecture design and training strategy of very deep networks. Earlier works adopted greedy layer-wise training or better initialization schemes to alleviate the vanishing gradients and diminishing feature reuse problems. The deeper version of our DANN also uses identity functions to construct skip connections.

The raw input video is first cut adaptively in the temporal domain and pre-defined in the spatial domain. The cropped video clip is then transformed via a Laplacian pyramid. Each scale is fed to the DANN which produces a set of feature maps. These feature maps at all scales are concatenated, the coarser scale maps being up sampled to match the size of the finest scale map. The Deep Alternative neural network is a deep network with alternating layers of recurrent networks.

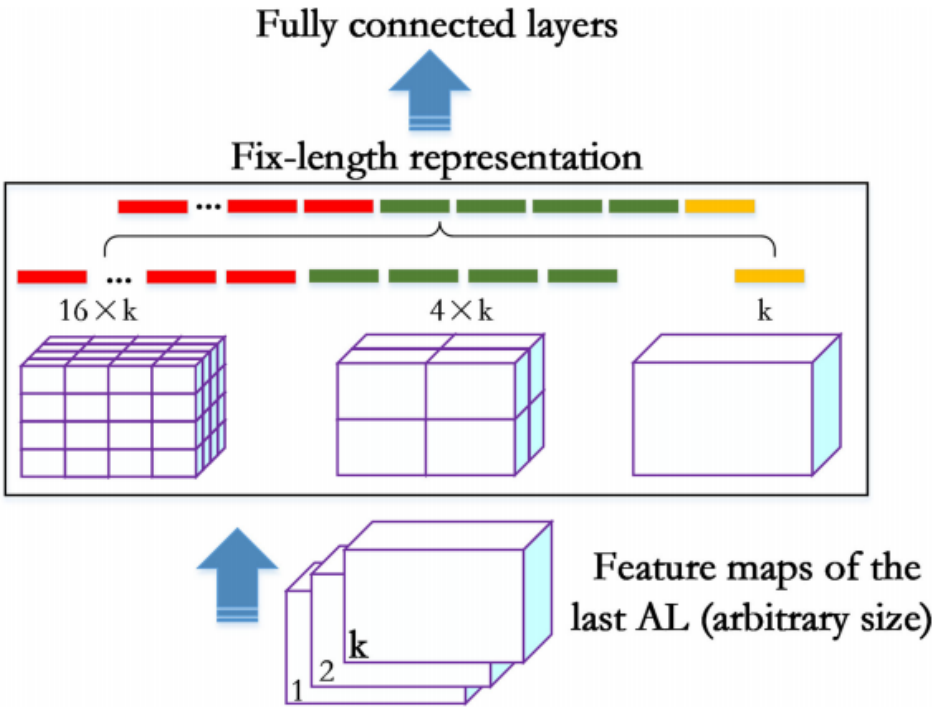




The recurrent connections in AL provide three major advantages compared with VCL, which is the standard module used in C3D networks. Many recent works have suggested that going deeper with convolutions often yields performance improvement. However, experiments suggest that simply adding ALs does not result in performance improvement. This is perhaps due to that current optimization techniques do not have sufficient power to optimize a large number of layers. In this subsection, we provide a strategy for going deeper using skip connections to release the full potential of the DANN. The proposed strategy is designed in a similar fashion as that of dropout but is performed vertically rather than horizontally.

In many videos, actions and events appear in various sizes. To capture this variability, our model should be scale-invariant. We extend our DANN to the multi-scale scenario, as shown in Figure 1. Our multi-scale DANN captures the invariant property by extending the concept of volumetric weight replication to the scale space. As mentioned above, weights are shared within the multiscale DANN. Intuitively, imposing complete weight sharing across scales is a natural way of forcing the network to learn scale-invariant features and, at the same time, reduces the chances of over-fitting. The more scales used to jointly train the models, the better the representation becomes for all scales.

The input size of a deep neural network for video classification in the temporal domain is often determined empirically since the evaluation all the choices is difficult in practice. Therefore the adaptive input size is made adaptive. . First, the local minima and maxima landmarks probably correspond to characteristic gesture and motion. Second, this signal is relatively robust with respect to changes in camera viewpoint, which are very common in realistic videos. One way to generate video tube features is to feed each tube into the DANN separately. However, this tends to be time-consuming since the computations needed for overlapping tubes are not shared. Spatial pyramid pooling improves upon the bag-of-word model in that it can maintain spatial information by pooling in local spatial bins. These spatial bins have sizes proportional to the image size; thus, the number of bins is fixed regardless of the image size. To adopt the DANN for input video clips of arbitrary sizes, we replace the last pooling layer with a novel VPPL inspired by the success of the spatial pyramid pooling layer.



Since DANN-6 and DANN-18 are special cases of the multiscale DANN, we describe here the training procedure of the multi-scale version. The training procedure for the other two structures is the same, although without two feed-forward steps, i.e., pyramid generation and feature map combination. To train a modern neural network for video classification, it is difficult to take the entire video as the input. . In the clip-wise training

procedure, for each video clip, the combined feature maps at all scales are sent to a softmax layer to obtain the probability of falling into the ‘e’th semantic category and that is used to train the model. The dataset is as such:

The results are as:

PERFORMANCE COMPARISON FOR DIFFERENT CONFIGURATIONS OF THE DANN ON THE UCF101 SPLIT 1 DATASET

Architecture	Clip	Video	Architecture	Clip	Video
B_6VCL_3FC	80.2	80.6	2AL_4VCL_VPP_3FC	85.9	85.1
AL_5VCL_VPP_3FC	85.1	86.2	3AL_3VCL_VPP_3FC	86.7	87.2
VC_AL_4VCL_VPP_3FC	83.3	84.1	4AL_2VCL_VPP_3FC	86.4	86.5
2VC_AL_3VCL_VPP_3FC	82.4	82.0	5AL_VCL_VPP_3FC	87.5	88.0
3VC_AL_2VCL_VPP_3FC	82.7	83.5	6AL_VPP_3FC	88.7	**89.0**
4VC_AL_VCL_VPP_3FC	81.4	81.9	6AL_VPP_3FC,3-layer LSTM	83.1	82.9
5VC_AL_VPP_3FC	80.9	81.5	6AL_VPP_3FC,5-layer LSTM	82.3	82.5

(a) Impact of the order and the number of AL using $\mathtt{T} = 3$ in DANN-6.

Architecture	Clip	Video
6AL_VPP_3FC, T = 2	87.5	88.5
6AL_VPP_3FC, T = 3	87.2	87.9
6AL_VPP_3FC, T = 4	87.5	88.5
6AL_VPP_3FC, T = 5	88.2	88.3
6AL_VPP_3FC, T = 6	**88.7**	**89.0**
6AL_VPP_3FC, T = 2	87.6	87.8
6AL_VPP_3FC, T = 3	87.3	87.3

(b) Impact of T in DANN-6 with all ALs.

Architecture	#Param.	Clip	Video
Wider-VCL-6	3.5M	79.2	78.3
Deeper-VCL-6	3.5M	78.8	78.7
DANN-6 (T = 0)	3.5M	81.2	80.3
DANN-6 (T = 1)	3.5M	83.1	83.5
DANN-6 (T = 6)	3.5M	88.7	89.0
DANN-18	10.1M	87.5	88.2
DANN-18 (skip)	10.1M	90.4	91.9
Multi-scale DANN-18	10.1M	92.2	92.8
Multi-scale DANN-18 + spatial	10.1M	**95.1**	**95.2**

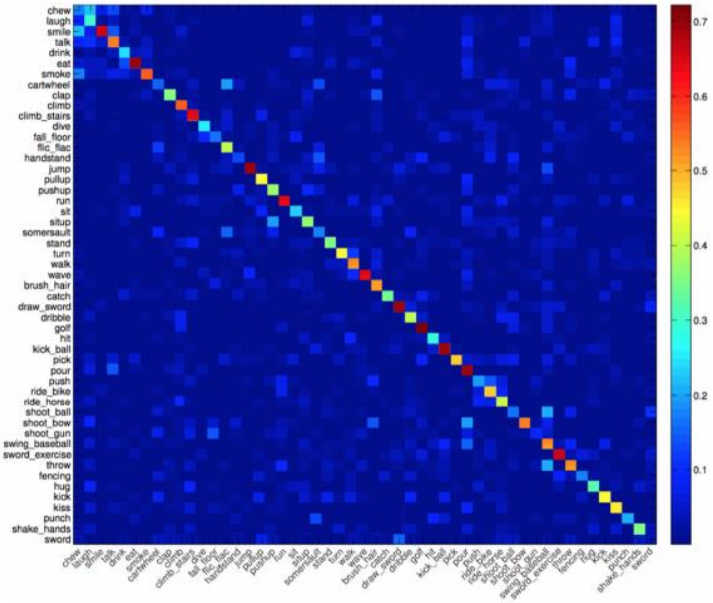
(c) Comparison with varying DANNs.

$\mathcal{S}$ \ p	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9	1.0
1	87.8	88.6	88.8	87.0	87.3	87.8	86.0	86.9	86.2	85.3
2	90.8	90.6	90.8	90.0	89.3	89.6	88.4	88.4	88.1	86.7
3	90.8	91.6	**92.8**	91.0	90.3	89.2	89.3	88.8	88.6	88.0
4	89.8	88.6	88.8	89.0	89.3	87.2	86.9	86.5	86.2	85.1
5	88.8	88.6	88.8	87.0	86.7	86.7	86.2	85.9	84.4	82.5

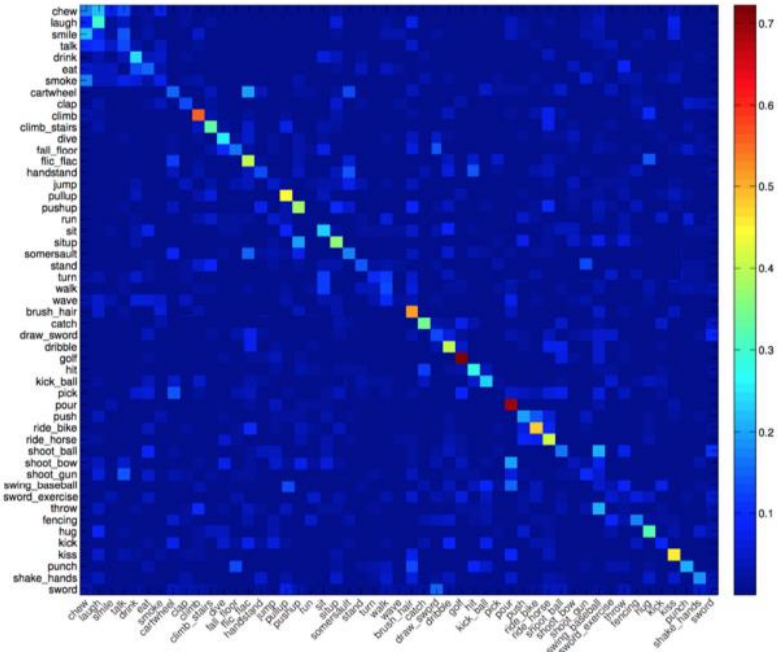
(d) Different probabilities of dropping each layer and scale numbers.

Method	Year	HMDB51	UCF101
Two-stream [10]	2014	59.4	88.0
Hidden Two-Stream [87]	2017	60.2	90.3
iDT+FV [18]	2013	57.2	85.9
iDT+HSV [3]	2016	61.1	88.0
MoFAP [88]	2016	61.7	88.3
LTC+spatial [14]	2016	61.5	88.6
KVMF [54]	2016	63.3	93.1
TDD+iDT [11]	2015	65.9	91.5
Two-Stream I3D [89]	2017	66.4	93.4
L^2STM [55]	2017	66.2	93.6
Adascan+iDT+C3D [46]	2016	66.9	93.2
SPN [51]	2017	68.9	94.6
ST-VLMPF [50]	2017	69.5	93.6
P3D ResNet + iDT [47]	2017	-	93.7
Cool-TSN [90]	2017	69.5	94.2
TSN [15]	2017	71.0	94.9
ShuttleNet [91]	2017	71.7	95.4
SMN+iDT [48]	2017	72.2	94.9
DOVF+MIFS [49]	2017	**75.0**	95.3
Wider-VCL-6		65.2	86.7
Deeper-VCL-6		66.8	88.3
DANN-6		69.3	89.2
DANN-18		71.9	93.6
Multi-scale DANN-18		73.9	95.5
Multi-scale DANN-18 + spatial		74.3	**95.7**

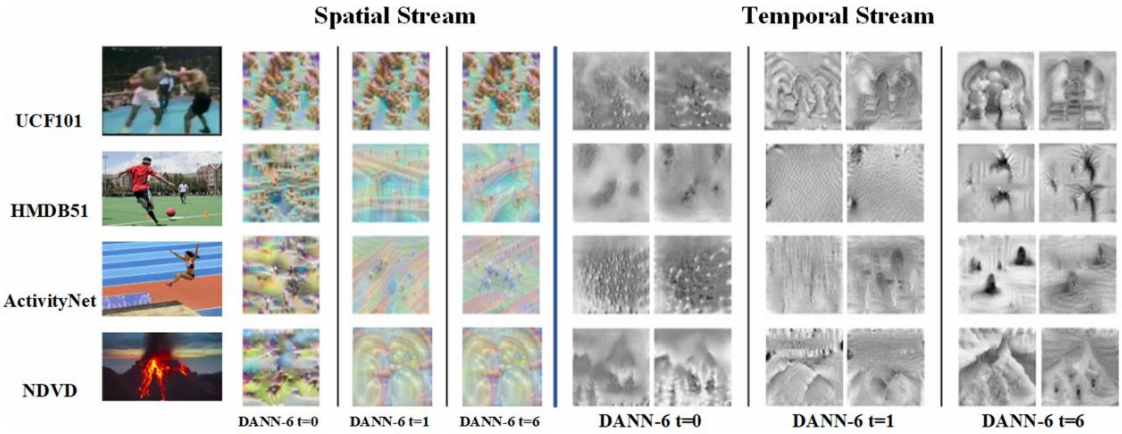
The confusion matrix for DANN on HMDB51 dataset is:



Confusion matrix for Deeper-VCL-6 on HMDB51 dataset:



Model visualization:



A novel deep alternative neural network for video classification, which enjoys the strength of both CNN and RNN by incorporating context evolutions into the forward propagation. We show how to go deeper with skip connection and construct a multi-scale version that facilitates the collection of rich context hierarchies. To evaluate our method beyond the realm of human-centric videos, we develop a new large-scale natural disaster video dataset and make it publicly available. The experiments on four challenging benchmarks demonstrate the advantages of our model on data related to both human actions and natural extreme events.

Paper 8: Automatic video genre classification

By: Kandasamy.K, Dr. Rajaram M.

Department of Electrical Engineering, Government college of Technology

The explosive growth of the video usage within the past two year is a direct result of the huge increase in internet bandwidth speeds. According to the Com Sore, a leading statistics reporting company for digital media, 150 million U.S. internet users have watched 96 videos in average per viewer in December 2008. This number is a resultant of the 13 percent increase of US online audience in the month of December 2008 compared to the previous month. The underline factor beneath this huge increase in online video usage is the surge in online traffic due to the increasing bandwidth speeds all across the globe. The automatic video classification problem is fundamentally different from the classical document classification problem. As mentioned earlier, this is mainly due to the semantic differences between a text file and a video file. In easiest terms, we can define a text file as a one-dimensional file, which contains only the text dimension. On the other hand, a video file can be defined as a three-dimensional file, which contains all three of the dimensions: audio, visual and text. There has been a significant progress in document classification research work using various different methods. It is not impossible to generate a video classification solution that will classify videos to different categories based on the content of that particular video. There has been a considerable amount of research and groundwork done by different people and organizations regarding the video classification problem. Automatic classification of digital video into various genres, or categories such as sports, news, commercials, and cartoons is an important task, and enables efficient cataloguing and retrieval with large video collections.

The best classification result now is with an accuracy of 95% using a dataset of 8 different genres. A video genre can be discriminated from the other based on the analogous features and attributes that is disparate from other genres. The feature is tested on four different video genres viz., cartoon, sports, commercial, and news. Today TRECVid has become a bench marking and evaluation cam pain for the automatic video classification field. However, most effort from TRECVid is focused on retrieving video information and using that information in a search query so that a user would be able to search through a video for a specific content.

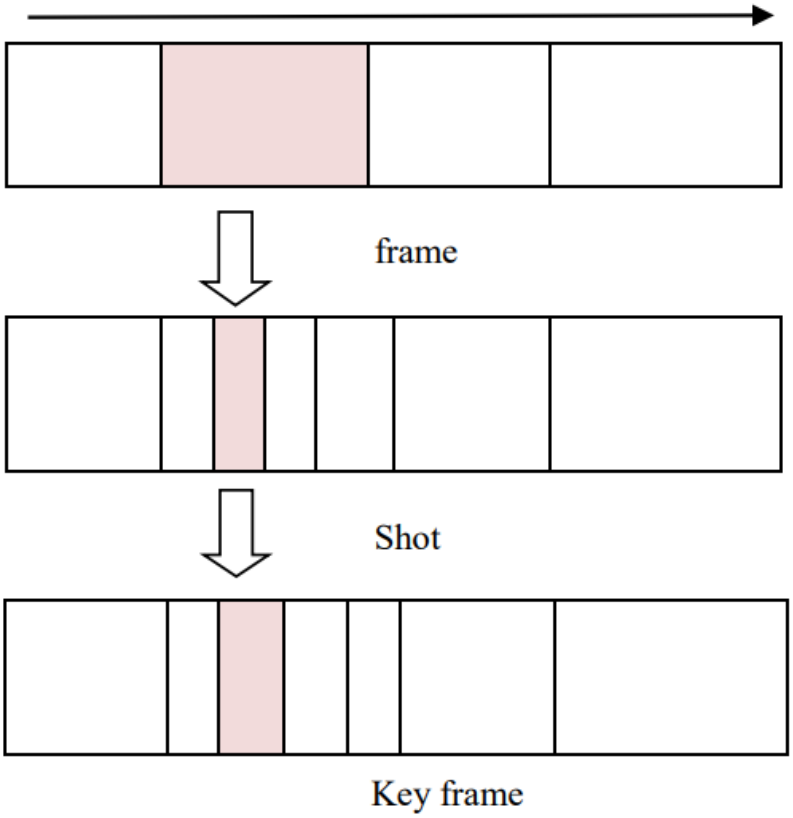
The automatic video classification problem is a slightly different problem when compared to the video retrieval problem. For instance, an automatic video classifier will define a particular video as a sports video or a news video, where as a video retrieval machine will focus on indexing each portion of the video for future retrieval.

Audio-based video classification is another form of categorization method that has gained a significant popularity when it comes to video classification technologies. Compared with pure text based classification, audio-based classification methods tend to yield results that are more accurate. Because of this reason, many video classification methods are based on audio-based approaches rather than text based approaches. Furthermore, compared to solely visual based approaches, audio-based approaches seem to in considerably lower processing cost. Based on the differences between audio and video files, audio-based approaches require less computational power than the visual based approaches. This frame-forming process is highly analogous to how visual scenes are processed. Additionally, to further enhance the process it is possible to collect these frames and define frame boundaries so that those frames can be identified using a key frame. After collecting these frames audio files are processed in two different domains: frequency domain and the time domain. Fourier Transform transfers the data from Time domain to Frequency domain. Silence and non-silence is first discriminated and then ZCR is used to classify them. Audio-based approaches are not the best way to classify videos but it certainly is a better approach than Text based classification.

Many video classification efforts that have been done so far is based on some kind of a visual based approach. This is very intuitive given that anyone would agree the visual element is the most important dimension out of the three dimensions of a video. Therefore, in order to gain from these visual clues most researchers have incorporated visual based approaches to their work. Most of these research that use visual features tend to extract features on a per frame or per shot basis.

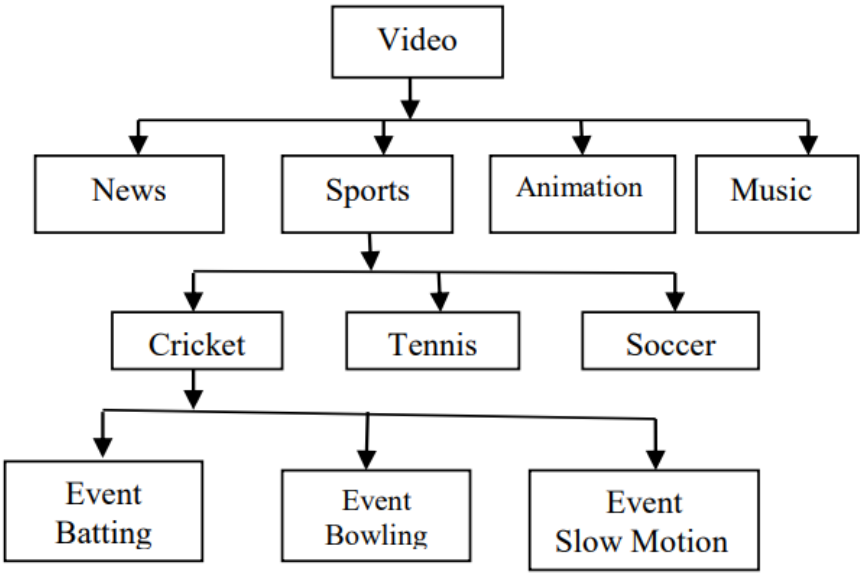
A video is a set of images in a sequential order. In addition, the shots are from a single camera. Therefore, each shot is separated in frames, features are recognized from each shot, and then the data is processed for training. Instead of analysing a video one frame at a time, it makes logical Sense to analyse it scene by scene. A whole scene can be represented as one logical unit of the whole story. For example in an action movie, there can be a scene where two people are fighting, or a car-chasing scene. The definition of a shot boundary can be different from one person to another based on multiple factors such as their taste, the way they analyse a clip, etc. Anyhow, we cannot undermine the importance of these shot based approaches. After all, it is one of the major factors that can contribute to a very good classification scheme.

Hybrid approaches are also considered for video classification. Most researchers have utilized at least one more combination of video, text and audio along with their primary choice to overcome unnecessary fluctuation of results. The entire classification method is based on how well this tracking system behaves. They have identified two issues involved in such object tracking methods: the detection of the targets in each frame and the extraction of object trajectories over frame sequences to capture their movements. For each detected face, the mean and the standard deviation in colour, the height, the width and the centre position have been computed. Authors have picked four types of TV programs to do their classification: news, commercials, sitcoms and soap operas. To classify a given video segment into one of these four categories, they have mapped it into the same feature space and evaluated its probability of being each category by the weighted distances to the centres of the news, commercial, sitcom and soap clusters. Key frame detection are the primary feature of this data extraction and learning process.



Broadcast video can be regarded as being made up of genre. The genre of a video is the broad class to which it may belong e.g., sports, cartoon, commercials etc., Genre can themselves in turn made up of genre. Genre

classifications at the same levels are manually exclusive. So the video genres can be regarded as a four structure. The various levels of video genre are shown.



Three types of colour-based classifications are used in this case and they are Bin colour histogram, pixel wise differentiation and motion recognition methods. Other than these, cartoon genre and edge feature are also dealt with.

Automatic classification using edge and colour histogram feature to identify the genre of video was achieved. Experiments are conducted on popular four video genres namely cartoon, sports, commercials and news. The approach initially evaluates the performance of two features individually by using cross correlation as the classifier method. Experiment shows that the approach is promising to classify the video genres. Finally, by combining the individual results the system gives a good classification accuracy of 94.5%.

Paper 9: Internet Video Traffic Classification Transfer across Video Streaming

By: D. J. Hani Mary Sheniha, A. B. Nivedha, and J. Shameera

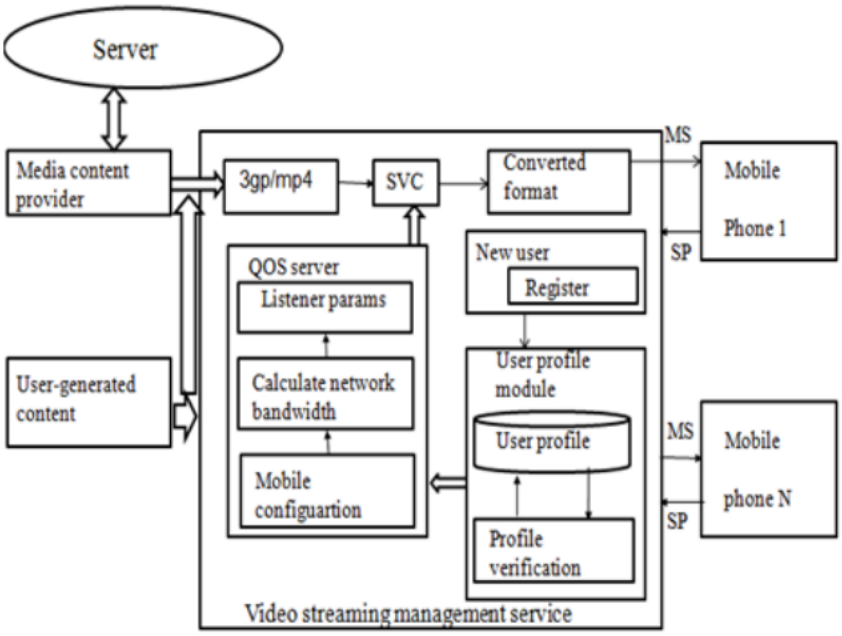
International Research Journal of Engineering and Technology (IRJET)
Volume: 05 Issue: 02 | Feb-2018

Driven by the hi-tech encroachment in wireless announcement systems, recent years have witnessed the marvellous expansion of mobile video traffic over the net. The reputation of influential mobile terminals promotes the astonishing sketch of mobile video services. As discharged within the current report IT people scrutiny firm Gartner, international data processor shipments has reached 2.4 billion in 2016, of those eighty-two are smart phones. Video streaming accounted for fifty fifth pace of the mobile traffic usage over and can reach seventy a decisive current trend linked this fabulous growth is that the quality of high definition (HD) video services. In spite of the speedy encroachment in mobile communication technologies, the network resources of existing 3G/4G systems are still constrained compared to the ever-growing video traffic.

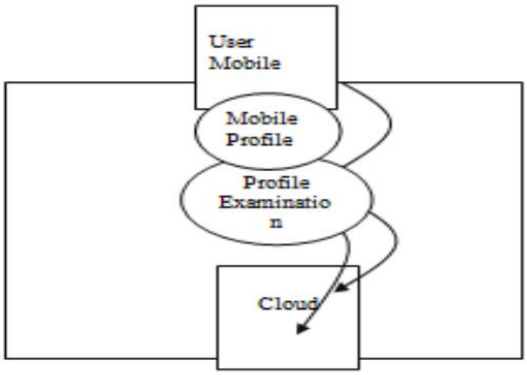
The video application insist on strict demand of instantaneity however copy toleration on video alteration. The vacillation and irresponsibleness of the communication methods in wireless networks, as one with the long delay, create crucial challenges to tackle the challenging Quality of expertise (QoE) requirements. We can understand there are two distinctive ways for multi-path data transfer: Multiple Access Point solutions (MAP), under such position there are more than one access points (for instance, a base station or an eNodeB acts as one access point, a WiFi AP act as another access point) for each smartphone. However, the smartphones are still contending transmission resources billed by these access points. Single Access Point solutions (SAP), in such circumstances there is only a single access point, which is usually the cellular base station. Each smartphone connects to this access point to battle restricted transmission resources using the cellular interface, while at the

same time using another interface (Blue Tooth or WiFi) to form a local network and carry out data sharing within this network. The suggestion of using multiple interfaces of mobile devices has been explored before but not in the same way as in this work. Chevrolet et al. presented a network layer architecture that enables various multi-access services. In addition, an algorithm called Earliest Delivery Path First (EDPF) is proposed, which ensures packets meet their playback deadlines by forecasting packets based on the anticipated liberation time of the packets. The authors in whispered that the end- to-end video frame delay was a brutally demanding problem for high definition online video services, and planned SFL, which is an original scheduling approach, and intentionally splits large-size video frames into sub-frames and dispatches each of them onto a different wireless network to the multi-homed client. However, the devices outfitted with multi-interfaces in the solutions above toughen their streaming capacity on their own and overlook the potential supportive opportunity, which may lead to the resource opposition on the inadequate wireless networks. Quite a lot of works discussed the usage of multi-interfaces in categorized delivery systems. Enchanting the social ties and geographical propinquity into account, painstaking a scenario in which device-to-device and cellular connections are used to broadcast the content. The authors proposed an infrastructure that exploits wireless multiplicity (channel diversity, network diversity, and technology diversity) to offer enhanced data presentation for wireless data users. The contemporary schemes always timetable the file data traffic in a content-agnostic approach without bearing in mind the complex video streaming personality such as the confident timeliness and understanding to delay and jitter requested by mobile video services.

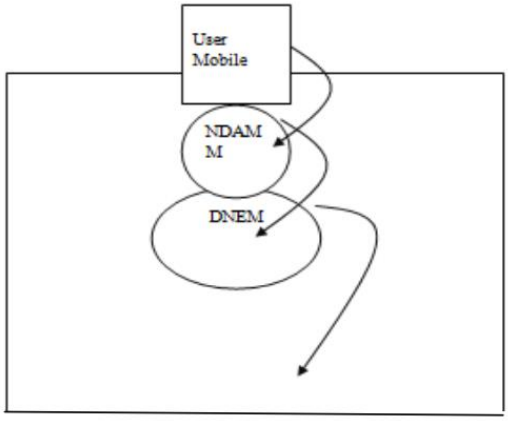
The proposed system architecture starts with user profile, web service connectivity, bandwidth estimation, video compression and multi-homed transmission. Initially user profile is created, then screen resolution for that user's device is estimated, based on that screen resolution uploaded videos can be transmitted to multi-homed devices. The schematic diagram is:



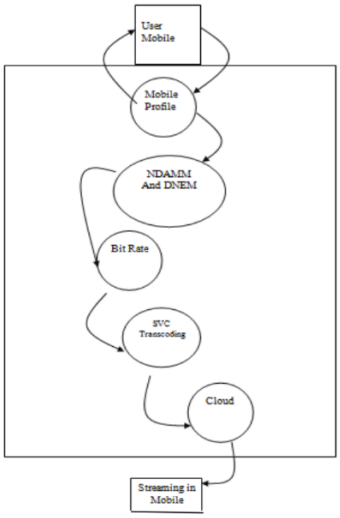
The profile agent is used to receive the mobile hardware environment parameters and create a user profile. Through a function, the mobile device can generate an XML-schema profile and transmitted to the profile agent. The schematic diagram is:



When web methods are invoked from inside Android application, the application gets back the data from the server in the form of XML. The response that has been received can be parsed and rendered in the application as needed by SOAP.

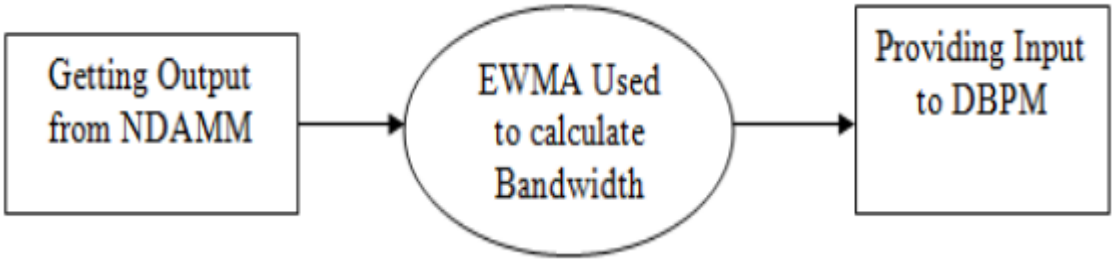


The NDAMM aims to determine the interactive communication frequency and the SVC multimedia file coding parameters according to the parameters of the mobile device. The architecture is as such:



It hands these over to the STC for Trans coding control, to reduce the communication bandwidth requirements and meet the mobile device user's demand for multimedia streaming. The DNEM is mainly based on the measurement based prediction concept; however, it further develops the Exponentially Weighted Moving Average (EWMA). If the network bandwidth value of this time cycle is within plus minus three standard

deviations of the standard value, the present mobile network will be in a stable state; otherwise, it will be in a fluctuating state. For the three video parameters of FPS, resolution and bit rate, the bit rate depends on the frame rate and resolution, so the Bayesian network adopts the frame rate and resolution as the video input features and uses the bit rate as parameter considered.



To conclude, the proposed system reduces the vulnerability of the mobile data theft from the user side. A set of adaptive networks and a device aware QoS approach for interactive mobile streaming was proposed. The experimental data proved that the method could maintain a certain level of multimedia service quality for dynamic network environment and ensure smooth and complete multimedia streaming services. Cloud services may accelerate research on SVC coding in the feature. The overall network environment and adjusting the interactive transmission frequency and dynamic multimedia Trans coding, to avoid the waste of bandwidth and terminal power.

Paper 10: A new video similarity measurement for sports video classification

By: Prisana Mutchima and Parinya Sanguansat

Rangsit Journal of Arts and Sciences
Submitted 8 August 2011; accepted in final form 27 November 2011

Many approaches have been attempted for video similarity measure and video classification. Following the literature review, one popular video representation technique is to represent each video sequence with frames, which contain all of the information of an image. In image comparison, various features such as colour, texture and shape were used in several approaches. Among these characteristics, colour features are the most basic features, which are widely used and prove to be highly effective for image comparison. A technique for video similarity measure based on the percentage of visually similar frames between the two sequences has been proposed. A common technique finds the total number of frames from each video sequence with at least one similar frame with the other sequence. Then, the ratio of these numbers will be computed to the total numbers of frames. After that, the threshold is used for comparing the difference between frames. The efficiency of such a technique depends on the effective selection of the optimal frame similarity threshold. Practically, it is rather difficult to identify the optimal frame similarity threshold because it often comes in an unpredictable pattern and has to be manually determined, resulting in time-consuming data processing.

The objective of this study is to efficiently measure video similarity by a new framework to reduce the dimensionality of video data by a random projection (RP) technique and fix dimension by a distance space technique. In addition, a compressive classification (CC) will be applied to classify videos. The data consist of 200 video sequences of TV sports programs, comprised of 10 sport genres, namely basketball, boxing, football, snooker, swimming, table tennis, tennis, beach volleyball, volleyball and wrestling. The datasets were divided into two groups, i.e. 100 training and 100 test video sequences. The number of frames of each video sequence is 30 frames per second in MPEG-2 format. The resolution of the datasets evaluation sequences is 480×720 pixels, and the length of each video is approximately 30 seconds.

For image classification, the colour histogram is widely used as an important colour feature indicating the content of the image. Moreover, the advantage of using the colour histogram is its robust ability for affine transformation, especially rotation and scaling of the image content. On the other hand, Naïve Video Similarity (NVS) is a traditional technique to measure video similarity. by finding the total number of frames from each

video sequence that has at least one visually similar frame with the other sequence, and then computing the ratio of this number to the overall total number of frames. High dimensional feature vectors represent individual frames in a video from a metric space. The final training and testing set can then be determined by a set of equation. Two such types of algorithms are described.

Algorithm 1 Distance Space Algorithm

Require: $X_{i=1\ldots N}, \xi_{k=1\ldots C}$

Ensure: $G_{i=1\ldots N}$

All frames of the training videos, X_i, are extracted by color histogram based method and projected by random matrix.

Perform clustering in this feature spaces to keep centroids of each cluster as reference vectors, ξ_k .

for $i = 1$ to N **do**

$G_i \leftarrow 0$

for $j = 1$ to $|X_i|$ **do**

for $k = 1$ to C **do**

$D_k \leftarrow \|X_i[j] - \xi_k\|$

end for

$L \leftarrow \arg\min_k (D_k)$

$G_i[L] \leftarrow G_i[L]+1$

end for

$G_i \leftarrow G_i / |X_i|$

end for

return $G_{i=1\ldots N}$

Algorithm 2 Sparse Classification Algorithm

Require: Test sample $v_{k,test}$, training matrix V, and error tolerance η.

Ensure: Estimated sparse weight α.

Solve the optimization problem expressed in Eq. (14).

For each class (i), repeat the following two steps.

a) Reconstruct a sample for each class by a linear combination of the training samples belonging to that class

using $v_{recon}(i) = \sum_{j=1}^{n_i} \alpha_{i,j} v_{i,j}$.

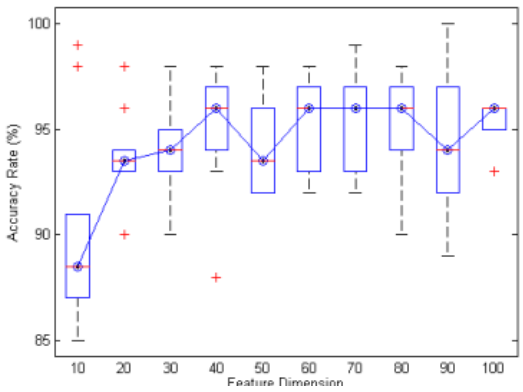
b) Find the error between the reconstructed sample and the given test sample by

$error(v_{test}, i) = \left\| v_{k,test} - v_{recon(i)} \right\|_2$.

Once the error for every class is obtained, choose the class having the minimum error as the class of the given test sample.

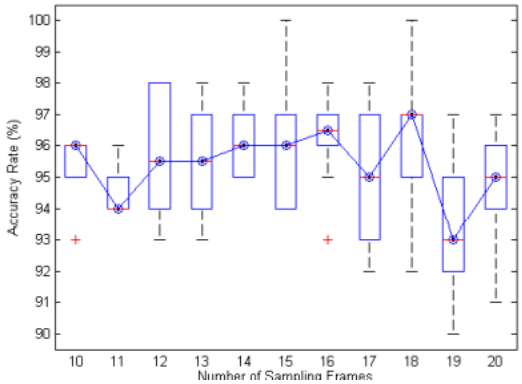
Mean and STD based on number of features:

Feature Dimension	Accuracy Rate	
	Mean (%)	S.D. (%)
10	90.00	4.83
20	93.50	2.42
30	94.20	2.30
40	95.00	2.94
50	94.00	2.16
60	95.30	2.36
70	***95.50***	***2.22***
80	95.30	2.63
90	94.30	3.13
100	95.40	0.97



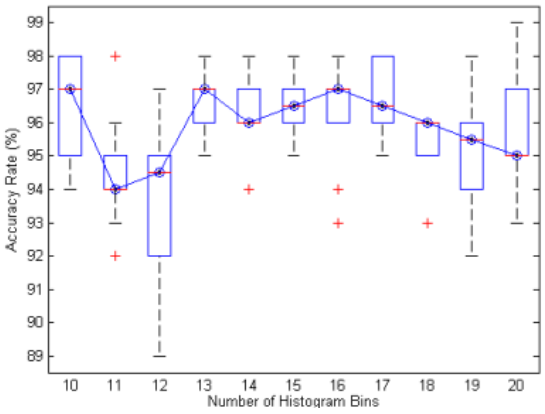
Mean and STD based on Number of sampling frames:

Number of Sampling Frames	Accuracy Rate	
	Mean (%)	S.D. (%)
10	95.40	0.97
11	94.60	0.84
12	95.60	1.90
13	95.60	1.71
14	96.30	1.16
15	96.00	1.94
16	96.30	1.49
17	94.80	2.20
18	***96.40***	***2.12***
19	93.30	2.21
20	95.00	1.76



Mean and STD based on histogram bins:

Number of Histogram Bins	Accuracy Rate	
	Mean (%)	S.D. (%)
10	96.40	1.58
11	94.50	1.65
12	93.70	2.45
13	***96.60***	***0.84***
14	96.00	1.25
15	96.50	1.08
16	96.30	1.57
17	***96.60***	***1.17***
18	95.40	0.97
19	95.00	2.00
20	95.70	1.77



Finally the overall accuracies:

Technique	Dimension	Accuracy Rate (%)
NVS Method	360,000	95.00
Expectation-based Method	360,000	97.00
Proposed Method	100	96.60

This paper proposes a new framework to enhance the performance in measuring video similarity and video classification. This framework applies the random projection (RP) technique to reduce the dimensionality of video data, and uses distance space techniques to fix video dimensions, followed by a compressive classifier to classify videos. This technique works with a dimensionality reduction method that is data independent. Moreover, when the number of dimension vectors becomes small, the classification process becomes quite fast. Thus, the proposed method can handle larger and longer videos. Comparing the efficiency of the NVS, the expectation-based and the proposed framework in video categorization, the results show that the dimension of the proposed framework is much smaller than other methods while the accuracy rate is comparable.

Paper 11: Video Classification using Machine Learning

By: Shaunak Deshpande, Ankur Kumar, Abhishek Vastrad, and Prof. Pankaj Kunekar

International Research Journal of Engineering and Technology (IRJET)
Volume: 07 Issue: 05 | May 2020

Human Behaviour Analysis (HBA) is a significant field of focus in artificial intelligence. It has many fields of use such as video monitoring, environment-assisted living, smart shopping environments, etc. The availability of human video data is increasing dramatically, with the help of leading companies in this area. From the perspective of DL this job approaches HBA. Owing to the growth of computing capacity, Deep Learning techniques have been a great step in the classification context in the last few years. Strategies used are Convolutional Neural Networks (CNNs) for image comprehension, and RNNs for temporary comprehension like video or text. The following sections discusses them in detail. Due to the better performance against conventional methods such as decision trees, support vector machines, Bayesian networks, etc., there is a growing interest in the last decade for the use of deep learning. Over the past few years, the increase in computing capacity has made it possible to consider the biologically inspired algorithms developed decades ago.

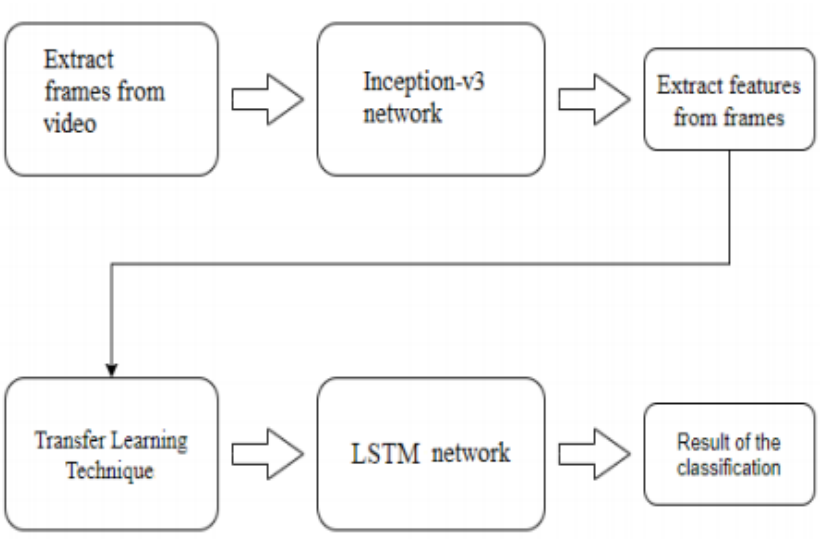
Due to the large amount of input data it would be very inefficient to remove the function with a normal Fully Connected (FC) network. In a broad sense, what CNN does is to minimize the details by looking at the individual regions of the data in order to extract specific features. CNNs are based on filters (kernels) that function like the weights of the Completely Connected ANN. Raising the size of the feature maps is a common technique for reducing the number of parameters and the amount of computation in a CNN. The pooling layer works independently (normally following a convolutional layer) and uses the maximum or average role in the feature map to reduce the regions. A very successful method for detecting features is the combination of many convolutional and pooling layers in parallel. It is because various kernel sizes can be implemented in parallel, allowing basic and complex functionality to be identified at different layers of the network.

The above-portrayed techniques are intended to arrange autonomous information, however what happens when we manage time-arrangement information? To handle with these necessities, another sort of neural system was intended to demonstrate time succession information. These systems are called Recurrent Neural Systems (RNNs) and permits the data to persist, by having loops in it .In this we get a streaming input xt, and thus a streaming output ot, in every iteration, the output ot, are another input for the subsequent iteration. Basic RNNs are useful to model small temporal dependencies. When coping with long sequences of information (in most real cases) a replacement variety of RNN called Long Short-Term Memory Networks are used.

LSTM networks, introduced by Hochreiter & Schmidhuber (1997), are RNN-based models capable of learning long-term dependencies. Vanilla RNNs have a very simple structure, like one perception with a singular tanh activation layer. In contrast, LSTM cells have a more complex structure, where rather than having one layer (tanh), there are four, interacting in an exceedingly very special way. The main purpose of this project is to design and implement an efficient deep learning solution able to predict and classify human behaviour into two categories, which are Safe Activity and Dangerous Activity by using combination of CNNs and RNNs architectures. Last decade was a prominent time for researchers to present many handcrafted and deep-net-based methods for action recognition. Earlier works were supported handcrafted features for non-realistic actions videos. Since the proposed method is predicated on deep neural network (DNN), during this section, we are going to only review related works supported DNN. In recent years, different variants of deep learning models are proposed for human action recognition in videos and have achieved great performance for computer vision tasks.

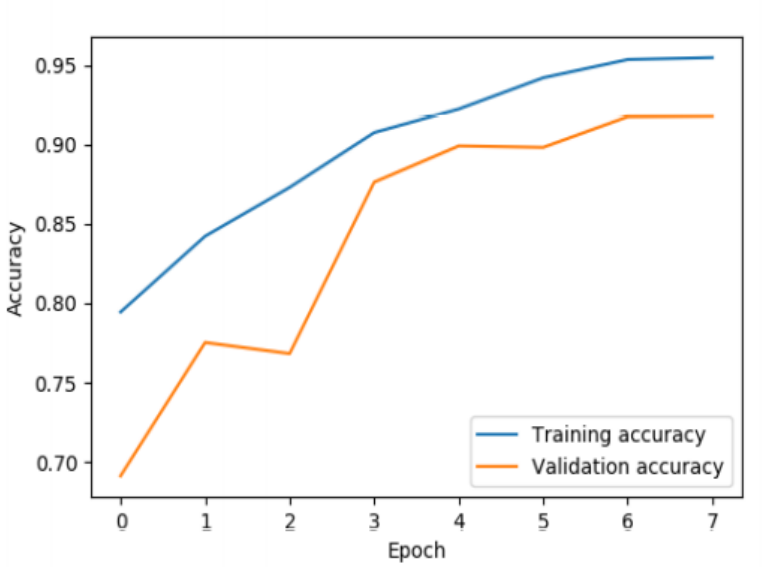
Upon evaluating the state-of-the-art of various behaviour recognition systems, identifying the methodology of the tools we will be using, and the expensive process of downloading the two data sets, we begin to introduce our proposal. This system is a combination of two separate DL models, a CNN reads the video frames and extracts the features, and RNN reads those features, predicting the operation. This DL model is scheduled in Python, using the Keras framework (using the Tensorflow framework as backend). In this method, the whole video is fed to the 3D CNN at once, and this CNN is capable of extracting not only image characteristics but also motion or time characteristics. All these features are then fed into a network of vanilla FCs. A common method in deep learning is to incorporate a pre-trained model to extract the features, and then then move the features on to the new model. Several models are pre-trained to recognize pictures. ImageNet is a database, which has been organizing an annual challenge (ILSVRC) since 2010, testing object detection and image classification algorithms. Several convolutions of various sizes are computed separately in these modules, and then concatenated into one row. This process allows extracting additional functionality. It also takes advantage of the 1 x 1 convolutions to reduce operations. The classification component is the second part of the Inception network, generated by a fully connected layer and a softmax output layer. This classification method is appropriate if we only need to classify an image at once, but if, we need to classify an image stream, such as a picture, and then we need RNNs.

RNNs that can model data sequences via internal loops that provide input to the network. Long Short-Term Memory Networks are a form of RNNs that can "remember" important parts of the input sequence, regardless of the time it shows up (simple RNNs only remember recent parts of the sequence, they have short term memories). In this model, it is suggested that an LSTM network adopt the function part of the network Inception. The model is:



As for implementation, Dataset building is the first procedure. The videos are acquired from YouTube and they are labelled. The next step is training. 75% of the data is used for training and 25% of it is used for testing. The labels are attached to the videos. In addition, these videos are broken down into frames, and these frames are then treated one by one, the data is extracted from these frames and they are pre-processed before training. The LSTM model contains two hidden layers with Relu and sigmoid activation respectively and output layer with two neurons with softmax activation to generate the classification for the video. Transfer learning is used for this reason. For testing, only one video is taken and that video is divided into frames. Features are extracted from these frames and stored into numpy arrays. Those features are then tested by model trained.

The result is:



The classification of video is troublesome because of numerous reasons, for example, shortage of video dataset, low accuracy etc. The main highlights of this paper are data manipulation of distinct datasets to fit a Deep Learning model; transfer learning of a pre-trained Deep Learning model (Inception V3) to our system; use of LSTM Recurrent Neural Networks. Since all day, every day manual checking of recordings is troublesome this framework could supplant such convention and analyse the footage with maximum accuracy.

Paper 12: Video event classification using string kernels

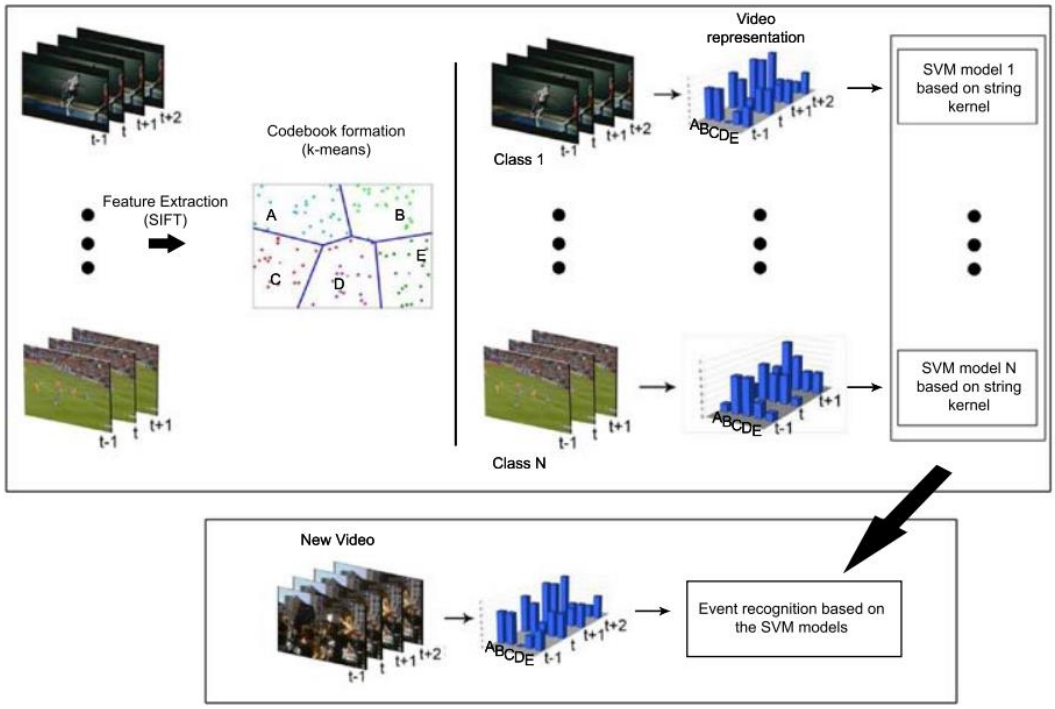
By: Lamberto Ballan, Marco Bertini, Alberto Del Bimbo, and Giuseppe Serra

Multimed Tools Appl
Published 15 September, 2019

Many approaches have been presented, but a common idea is to model a complex object or a scene by a collection of local interest points. Each of these local features describes a small region around the interest point and therefore they are robust against occlusion and clutter. To achieve robustness to changes of viewing conditions the features should be invariant to geometrical transformations such as translation, rotation, scaling and affine transformations. SIFT is considered to be the primary feature extracting function over here. . In the visual domain, an image or a frame of a video is the visual analogue of a document and it can be represented by a bag of quantized invariant local descriptors (usually SIFT), called visual-words. The main reason for its success is that it provides methods that are sufficiently generic to cope with many object types simultaneously. . Even if few novel Spatio-temporal features have been proposed, the most common solution is to apply the traditional BoW approach using static features (e.g. SIFT) on a key frame basis. Unfortunately, for this purpose the standard BoW approach has shown some drawbacks with respect to the traditional image categorization task. An action is described by a “phrase” of variable length, depending on the clip’s duration, thus providing a global description of the video content that is able to incorporate temporal relations.

Experiments have been performed on soccer and news video datasets, comparing the proposed method to a baseline KNN classifier and to a traditional key frame-based BoW approach. Experimental results obtained by SVM and string kernels outperform the other approaches and, more generally, they demonstrate the validity of the proposed method. Video event detection and recognition is really challenging because of complex motion, occlusions, clutter, geometric transformations and illumination changes. Nevertheless, it is an essential task for automatic video content analysis and annotation. Previous works in this field can be roughly grouped into three main categories. Over the past decade, the specific problem of recognizing human actions has received considerable attention from the research community. In fact, an automatic human activity recognition method may be very useful for many applications such as video surveillance, video annotation and retrieval and human–computer interaction.

Structurally an event is represented by a sequence of frames, that may have different lengths depending on how it has been carried out. We model an event by a sequence of visual word frequency vectors, computed from the frames of the sequence; considering each frequency vector as a character we call this sequence (i.e. string) phrase. Additionally, we define a kernel, based on an edit-distance, used by SVMs to handle variable-length input data such as this kind of event representation.



As previously introduced, each video shot is described as a phrase (string) formed by the concatenation of the bag-of-words representations of consecutive characters (frames). To compare these phrases, and consequently actions and events, we can adapt metrics defined in the information theory. The edit distance between two string of characters is the number of operations required to transform one of them into the other.

The Video representation is shown as:

(a) text example

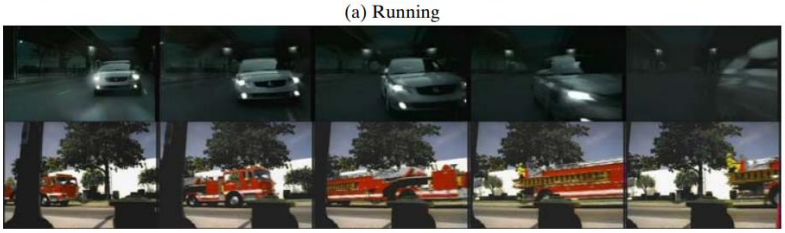
		S	E	N	D
	0	1	2	3	4
A	1	1	2	3	4
N	2	2	2	2	3
D	3	3	3	3	2

(b) video example

						
	0	1	2	3	4	5
	1	0	1	2	3	4
	2	1	1	2	2	3
	3	2	1	1	2	3
	4	3	2	2	2	2

The dataset used for training are as such:

(a) Running



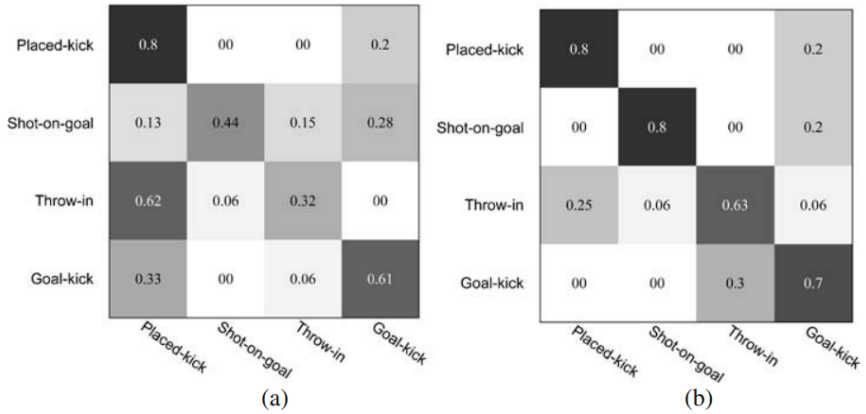
(b) Exiting Car

A set of different tests and experiments are performed over time to check the features extracted from these. In recent years, Support Vector Machines (SVMs), introduced, have become an extremely popular tool for solving classification problems. In their simplest version, given a set of labelled training vectors of two classes, SVMs map these vectors in a high dimensional space and learn a linear decision boundary between the two classes that maximizes the margin, which is defined to be the smallest distance between the decision boundary and any of the input samples. The result is a linear classifier that can be used to classify new input data.

In order to evaluate results, the following metrics were used:

Metric	Th	Accuracy
Bhattacharyya	0.5	0.47
Chi-square	**0.13**	**0.54**
Correlation	0.7	0.53
Intersection	0.1	0.52
Kolmogorov-Smirnov	0.5	0.50
Mahalanobis	7	0.37

The results of KNN and SVM classifiers:



	kNN	SVM
Mean Accuracy	0.54	**0.73**

(c)

The results show that SVM with string kernels outperform both the performance of the baseline kNN classifiers and of the standard BoW approach and, more generally, they exhibit the validity of the proposed method. Our future work will deal with the application of this method to a broader set of events and actions that are part of the TRECVID LSCOM events/activities list, and the use of other string kernels. Moreover, we will investigate the possibility to integrate the proposed approach in an ontology-based framework that exploits concept and event dependencies to improve the quality of classification.

Paper 13: Contour based classification of video objects

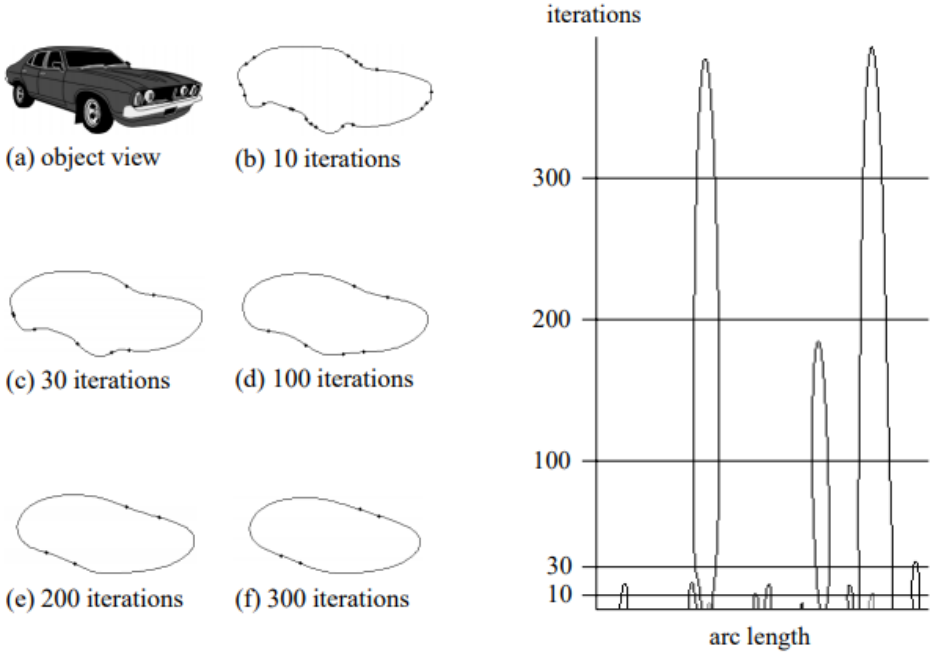
By: Stephan Richter, Gerald Kuhne and Oliver Schuster

University of Mannheim, Germany

The recognition of objects that appear in a video sequence constitutes another essential part of any video content analysis system. In general, object recognition can be addressed at different levels of abstraction. For instance, an object might be classifiable as a “cat” (object class), as a “Siamese cat” (subordinate level) or as “my neighbour’s cat” (individual object).

Contour-analysis techniques have existed in computer science for some time now. Pavlidis published one of the first overviews of algorithms in the area of shape analysis as early as 1978. He restricted his review to the analysis of “silhouettes”, as he called shapes and contours of two-dimensional objects. Already in 1978, Pavlidis mentioned that shape analysis is “an enormous subject”. More than twenty years later, the subject is still enormous, many new approaches have been tried, and some progress has been made. Pavlidis mentioned that it seemed to be possible to develop rigorous mathematical algorithms to analyse shapes and provide results similar to human perception. Our research found that no straightforward mathematical metric could be found which models human perception with regard to shape analysis. One of the more promising contour analysis techniques is the CSS method introduced by Mokhtarian. The advantages of his method are that it is size and rotation invariant and robust to noise. In the next section, we briefly describe the CSS method and outline the architecture of our video object recognition system.

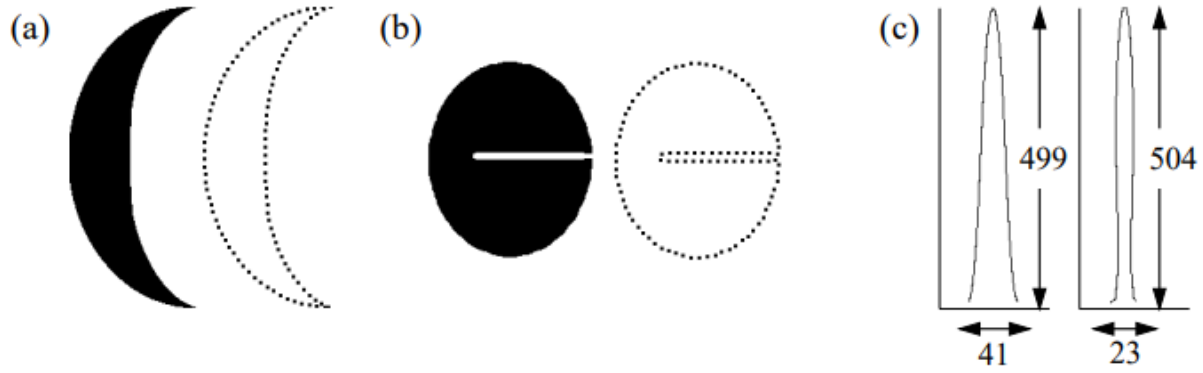
With every passing iteration, the contours keep on smoothening. It is shown below:



Different sources of information are available to characterize a two-dimensional view (e.g. contour, colour, texture, motion, or relative location of the object to other objects). However, most common objects can be identified by their contours only. 22 In our approach, for each object view a few parameters are extracted from

its contour and stored in conjunction with the object view's class name in a database. The parameters are calculated using a modified curvature scale space (CSS) technique.

Example of types of concavity:



From the positions of the zero crossings at different scales, a so-called CSS image is constructed. The CSS image shows the zero crossings with respect to their position on the contour and the width of the Gaussian kernel (or the number of iterations see Figure 2). Therefore, significant contour properties that are visible for a large number of iterations result in high peaks in the CSS image. However, areas with rapidly changing curvatures caused by noise produce only small local maxima. The shallow concavity of the object contour shown in (a) and the deep concavity of the object contour displayed in (b) result in peaks of nearly the same height (relative difference about 1 %) in the CSS images (c). Consequently, certain contours differing significantly in their visual appearance are claimed by the basic CSS technique to be similar.

Object matching is done in two steps. In the first, each individual object in a sequence is compared to all objects in the database by comparing peaks characterised by the triplets in the database. A list of the best matches is built for further processing. It might be necessary to rotate or mirror one of the images so that the peaks are aligned best. Next, a matching peak is determined for each peak in cm1. If a matching peak is found, the Euclidean distance of the height and position of the peaks is calculated and added to the difference between the images. If no matching peak was found, the height of the peak in cm1 multiplied by a penalty factor is added to the total difference.

Next is Sequence based matching, this means that the frames, arranged sequentially are used to extract features from and then trained sequentially. Once the matching algorithm has been executed, a list of matches for each object in the sequence exists. This list contains the difference to the object view in the database and the object class of the object view. Only the top match, i.e. the object view with the least difference, is used for evaluation.

The results are as such:

Sequence	Segmentation	Number of frames	Number of frames matched	Object class detected	
People–1	automatically	26	21	People	96 %
People–2	automatically	39	38	People	96 %
People–3	automatically	39	23	People	44 %
People–4	manually	29	26	People	71 %
People–5	manually	13	6	People	67 %
Bird–1	manually	15	10	People	57 %
Car–1	manually	51	33	Cars	100 %
Car–2	manually	21	11	Cars	87 %
Car–3	manually	51	40	Cars	100 %
Car–4	manually	19	14	Cars	100 %
Car–5	manually	22	17	Cars	100 %

In all human sequences the object class people is the one with the most matches. The number of top matches ranges from 44 % to 96 %. The recognition of objects appearing in video sequences is one of the most challenging tasks in the field of automatic video content analysis. We presented a system of object classification that relies on a database containing pre-processed two-dimensional views of prototypical video objects. Classification is performed by matching curvature features of the presegmented video object in question to object view representations in the database.

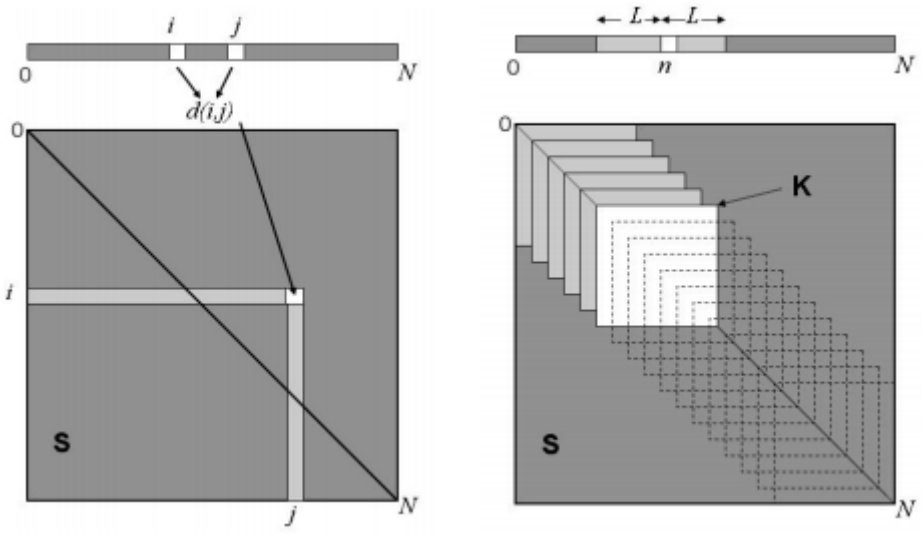
Paper 14: Video Segmentation via Temporal Pattern Classification

By: Matthew Cooper, Ting Liu, and Eleanor Rieffel

IEEE Transactions on Multimedia,
November 2008

A great deal of current video analysis research focuses on automatically extracting semantics from multimedia within the broader context of multimedia information retrieval. Semantic video annotation and video retrieval are also the focus of the highly successful TRECVID evaluations. Shot boundary detection is a core task at TRECVID, and shots serve as the units for both higher-level semantic annotation and retrieval tasks.

The similarity matrix embedding and kernel correlation are depicted as follows:



Abrupt shot boundaries exhibit a distinct pattern in the similarity matrix. Frames in visually coherent shots have low within-shot dissimilarity (high similarity). Frames from two such shots that are adjacent in time generally show high between-shot dissimilarity (low similarity). This produces a checkerboard along the main diagonal of S whose crux is the diagonal element corresponding to the boundary frame. This observation has motivated matched filter approaches to boundary detection, which we refer to as kernel correlation. The matched filter is a square kernel matrix, K, that represents the appearance of an ideal boundary in S.

A system using Kernel correlation is defined. This is defined for certain kernels only. The kernel compares current frame with previous frame and vice versa. The elements containing unfilled circles contribute negatively to the novelty score. As for the other one, statistical boundary detection is taken into consideration and here just the boundaries of objects are detected. This work used a Poisson prior for shot duration, and parametric templates for the class conditional models for discontinuity (dissimilarity) features. It also included a "detector cascade" approach, similar to that employed in our system, to handle detection of multiple types of shot transitions.

Several interesting high performance systems were presented in the 2004 TRECVID evaluation [2], including a version of the system presented in this paper use a system processing MPEG features directly with motion compensation to improve the temporal resolution. They use pixel and edge-based low-level features and detectors for specific gradual transition types such as wipes and dissolves, as well as a (camera) flash detector.

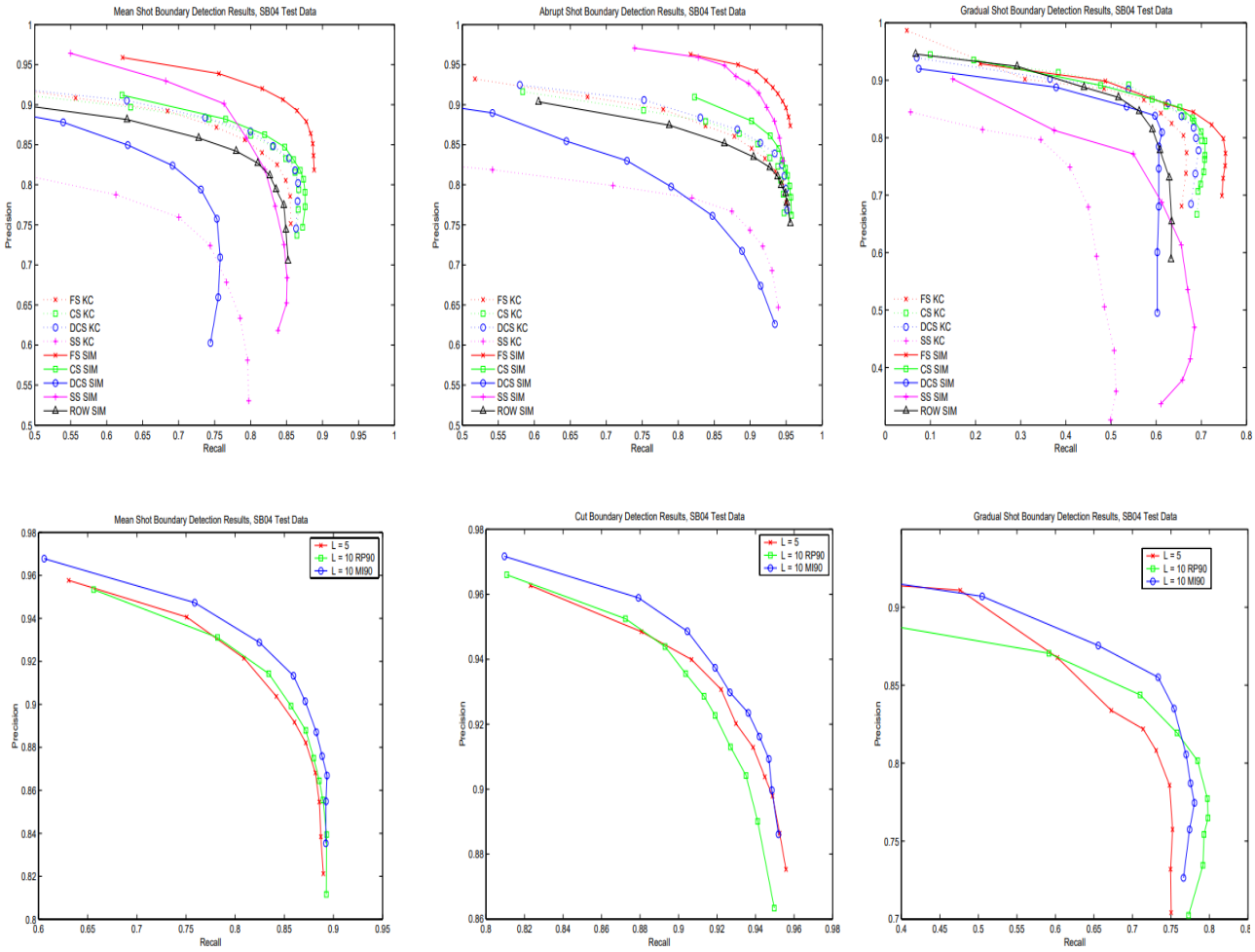
This system represents a basic framework in which any combination of low-level features and similarity measures can be used. Additionally, the approach can be adapted to any other modality or time-ordered data collection.

Low-key features are detected by building intermediate features. YUV colour histograms are used, which are a simple and common feature parametrization. We compute 32-bin global frame histograms, and 8-bin block histograms using a 4 × 4 uniform spatial grid for each channel. Elements are eliminated from the main diagonal (always zero) and remove duplicates due to the symmetry of the similarity matrices.

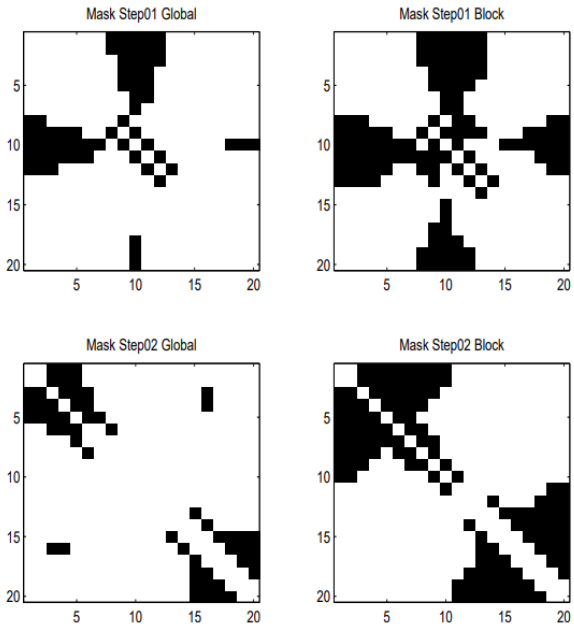
The intermediate features are classified to detect shot boundaries using a binary KNN classifier. The KNN classifier has two appealing properties. First, it is non-parametric, making no limiting assumptions about the statistics of the transition classes expressed in our intermediate features. Secondly, the asymptotic error rate of the 1-nearestneighbor classifier does not exceed twice the Bayes (minimum) rate. To reduce complexity, a number of efficient implementations have been devised. The accelerated version we employ uses metric tree data structures to achieve efficient spatial search.

Information-theoretic feature selection is done here at very first and detection of two types of shot boundaries: abrupt (cut) and gradual. Intuitively, we expect that specific inter-frame comparisons will be of varying relevance to detecting these two classes. The greedy approach ignores inter-feature redundancies, and the resulting feature subset is not generally maximally informative. To account for inter-feature dependencies, more measures that are complicated must be calculated.

The results are as follows:



The figure depicts the kernels selected using the greedy information theoretic procedure.



We conclude from the previous section that building intermediate features corresponding to the FS kernel provides excellent overall performance. The performance of the various systems demonstrate that adding

information to the intermediate features generally benefits performance. At the same time, larger intermediate feature vectors increase the computational requirements of classification. The goal of this section is to explore this trade-off between performance and complexity using feature selection. The block histogram comparisons are better represented in all subsets. Visually, the feature subsets in the first and second classification steps are largely disjoint. This supports the use of feature selection to reduce complexity by separately optimizing the two steps.

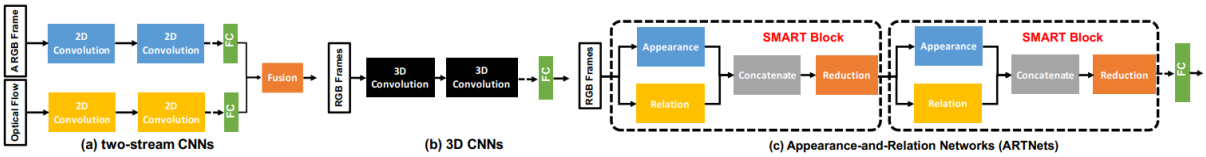
There are several possible directions for future work. Algorithmic efficiency could be significantly improved by adapting the method to operate directly on compressed streams, or by developing schemes to avoid sequentially processing every frame. This will necessarily incur a performance loss, but will also reduce complexity. More complete analysis of inter-feature redundancies in the framework of can also be performed. Since the feature selection step is off-line, the complexity of this analysis will not impact the system at run time, but can be expected to provide further improvements in performance. Finally, we believe the method can be applied to other modalities such as text and audio, and combinations of these modalities, given appropriate measures of similarity and low-level features.

Paper 15: Appearance-and-Relation Networks for Video Classification

By: Limin Wang, Wei Li, Wen Li, and Luc Van Gool

1National Key Laboratory for Novel Software Technology, Nanjing University, China

Deep learning has witnessed a series of remarkable successes in computer vision. In particular, Convolutional Neural Networks (CNNs) have turned out to be effective for visual tasks in image domain, such as image classification, object detection and semantic segmentation. Deep models have been also introduced into video domain for action recognition, and obtain comparable or better recognition accuracy to those traditional methods with handcrafted representations. There are three kinds of successful architectures or frameworks for video classification: (1) two-stream CNNs, (2) 3D CNNs, and (3) 2D CNNs with temporal models on top such as LSTM, temporal convolution, sparse sampling and aggregation, and attention modelling. Two-stream CNNs capture appearance and motion information with different streams, which turn out to be effective for video classification. Our SMART block aims to Simultaneously Model Appearance and RelaTion from RGB input in a separate and explicit way with a two-branch unit, in contrast to modelling them with two-stream inputs.

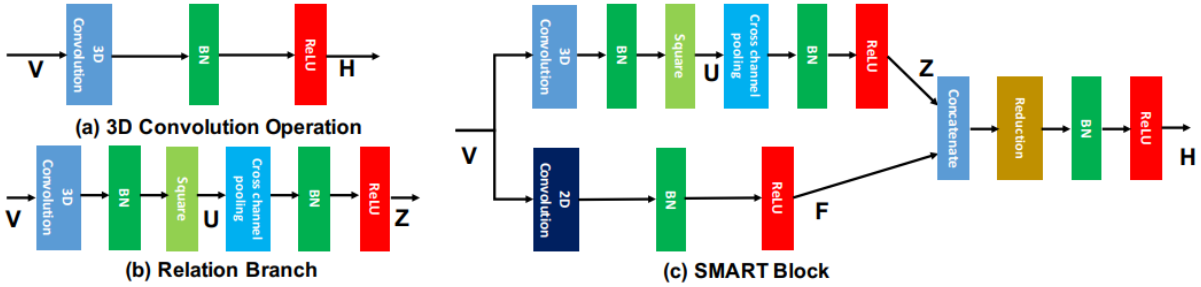


ARTNet is tested on the task of action recognition in video classification. Particularly, we first study the performance of the ARTNet on the Kinetics dataset [20]. We observe that our ARTNet obtains an evident improvement over C3D, and superior performance to the exiting state-of-art methods on this challenging benchmark under the setting of training from scratch with only RGB input. ARTNet are empirically investigated on the large-scale Kinetics benchmark and state-of-the-art performance on this dataset is obtained under the setting of using only RGB input and training from scratch.

Our work focuses on short-term temporal modelling and is most related with 3D CNNs. Our ARTNet mainly differs to 3D CNNs in that we design a new SMART block to model appearance and relation separately and explicitly with a two-branch architecture, while 3D CNNs employ the 3D convolutions to capture appearance and relation jointly and implicitly.

First, we discuss the role of multiplicative interaction in modelling relation across multiple frames. Next, we introduce the design of a SMART block. Finally, we propose the ART-Net by stacking multiple SMART blocks in the architecture of C3D-ResNet18. A natural solution to this problem is to perform standard feature learning on the concatenation of these two patches, just like a 3D convolution. Thus, this solution couples the information of appearance and relation, adding the modelling difficulty and increasing the over-fitting risk. Assuming the

independence between appearance and relation, it is reasonable to decouple these two kinds of information when designing learning modules. Factorizing the parameter tensor W into three matrices would be an efficient way to reduce model parameters.



The branches of these networks are defined in a usual manner, and the following table shows the three types of network architectures. 2 types of training blocks 3D convolutions and SMART blocks are inserted in this.

layer name	output size	C3D-ResNet18	ARTNet-ResNet18 (s)	ARTNet-ResNet18 (d)
conv1	$56 \times 56 \times 8$	3D conv $7 \times 7 \times 3$, stride $2 \times 2 \times 2$	SMART $7 \times 7 \times 3$, stride $2 \times 2 \times 2$	
conv2_x	$56 \times 56 \times 8$	$\begin{bmatrix} \text{3D conv} & 3 \times 3 \times 3 & 64 \\ \text{3D conv} & 3 \times 3 \times 3 & 64 \end{bmatrix} \times 2$	$\begin{bmatrix} \text{3D conv} & 3 \times 3 \times 3 & 64 \\ \text{3D conv} & 3 \times 3 \times 3 & 64 \end{bmatrix} \times 2$	$\begin{bmatrix} \text{3D conv} & 3 \times 3 \times 3 & 64 \\ \text{SMART} & 3 \times 3 \times 3 & 64 \end{bmatrix} \times 2$
conv3_x	$28 \times 28 \times 4$	$\begin{bmatrix} \text{3D conv} & 3 \times 3 \times 3 & 128 \\ \text{3D conv} & 3 \times 3 \times 3 & 128 \end{bmatrix} \times 2$	$\begin{bmatrix} \text{3D conv} & 3 \times 3 \times 3 & 128 \\ \text{3D conv} & 3 \times 3 \times 3 & 128 \end{bmatrix} \times 2$	$\begin{bmatrix} \text{3D conv} & 3 \times 3 \times 3 & 128 \\ \text{SMART} & 3 \times 3 \times 3 & 128 \end{bmatrix} \times 2$
conv4_x	$14 \times 14 \times 2$	$\begin{bmatrix} \text{3D conv} & 3 \times 3 \times 3 & 256 \\ \text{3D conv} & 3 \times 3 \times 3 & 256 \end{bmatrix} \times 2$	$\begin{bmatrix} \text{3D conv} & 3 \times 3 \times 3 & 256 \\ \text{3D conv} & 3 \times 3 \times 3 & 256 \end{bmatrix} \times 2$	$\begin{bmatrix} \text{3D conv} & 3 \times 3 \times 3 & 256 \\ \text{SMART} & 3 \times 3 \times 3 & 256 \end{bmatrix} \times 2$
conv5_x	$7 \times 7 \times 1$	$\begin{bmatrix} \text{3D conv} & 3 \times 3 \times 3 & 512 \\ \text{3D conv} & 3 \times 3 \times 3 & 512 \end{bmatrix} \times 2$	$\begin{bmatrix} \text{3D conv} & 3 \times 3 \times 3 & 512 \\ \text{3D conv} & 3 \times 3 \times 3 & 512 \end{bmatrix} \times 2$	$\begin{bmatrix} \text{3D conv} & 3 \times 3 \times 3 & 512 \\ \text{3D conv} & 3 \times 3 \times 3 & 512 \end{bmatrix} \times 2$
	$1 \times 1 \times 1$	average pool, dropout, 400-d fc, softmax		
params (M)	-	33.37	33.39	35.20
FLOPs (G)	-	19.58	19.97	23.70

The network parameters are initialized randomly. We use the mini-batch stochastic gradient descent algorithm to learn network parameters, where the batch size is set to 256 and momentum is set to 0.9. The frames are resized to 128×170 and then a volume of 112×112×16 is randomly trimmed and cropped from each training video. This volume also undergoes a random horizontal flip, with the per-pixel mean subtracted.

The learning rate is initialized as 0.1 and divided by a factor of 10 when validation loss saturates. The total number of iteration is 250, 000 on the Kinetics dataset. To reduce the risk of over-fitting, we add a dropout layer before the final classification layer, where the dropout ratio is set to 0.2. For testing network, we follow the common evaluation scheme, where we sample 250 volumes of 112 × 112 × 16 from the whole video. Specifically, we first uniformly trim 25 clips of 128×170×16 and then generate 10 crops of 112×112×16 from each clip (4 corners, 1 centre, and their horizontal flipping). The final prediction result is obtained by taking an average over these 250 volumes.

The results are as such:

Method	Spatial resolution	Backbone architecture	Kinetics val set	Kinetics test set
ConvNet+LSTM [5, 33]	299 × 299	ResNet-50	-	68.0%
Two Stream Spatial Networks [35]	299 × 299	ResNet-50	-	66.6%
C3D [41]	112 × 112	VGGNet-11	-	67.8%
C3D [42]	112 × 112	ResNet-18	75.7%	74.4%
C3D [42]	112 × 112	ResNet-34	77.0%	75.3%
TSN Spatial Networks [49]	224 × 224	Inception V2	77.8%	-
RGB-I3D [2]	224 × 224	Inception V1	-	78.2%
ARTNet w/o TSN	112 × 112	ResNet-18	78.7%	77.3%
ARTNet with TSN	112 × 112	ResNet-18	**80.0%**	**78.7%**

Method	Pre-train dataset	Spatial resolution	Backbone architecture	UCF101	HMDB51
HOG [44]	None	240 × 320	None	72.4%	40.2%
ConvNet+LSTM [5]	ImageNet	224 × 224	AlexNet	68.2%	-
Two Stream Spatial Network [35]	ImageNet	224 × 224	VGG-M	73.0%	40.5%
Conv Pooling Spatial Network [8]	ImageNet	224 × 224	VGGNet-16	82.6%	-
Spatial Stream ResNet [7]	ImageNet	224 × 224	ResNet-50	82.3%	43.4%
Spatial TDD [46]	ImageNet	224 × 224	VGG-M	82.8%	50.0%
RGB-I3D [2]	ImageNet	224 × 224	Inception V1	84.5%	49.8%
TSN Spatial Network [49]	ImageNet	224 × 224	Inception V2	**86.4%**	**53.7%**
Slow Fusion [19]	Sports-1M	170 × 170	AlexNet	65.4%	-
C3D [41]	Sports-1M	112 × 112	VGGNet-11	82.3%	51.6%
LTC [43]	Sports-1M	71 × 71	VGGNet-11	82.4%	48.7%
C3D [42]	Sports-1M	112 × 112	ResNet-18	**85.8%**	**54.9%**
TSN Spatial Network [49]	ImageNet+Kinetics	224 × 224	Inception V2	91.1%	-
TSN Spatial Network [49]	ImageNet+Kinetics	229 × 229	Inception V3	93.2%	-
RGB-I3D [2]	ImageNet+Kinetics	224 × 224	Inception V1	**95.6%**	**74.8%**
C3D	Kinetics	112 × 112	ResNet-18	89.8%	62.1%
ARTNet w/o TSN	Kinetics	112 × 112	ResNet-18	93.5%	67.6%
ARTNet with TSN	Kinetics	112 × 112	ResNet-18	**94.3%**	**70.9%**

In the paper it has been presented a new architecture, coined as ARTNet, for spatiotemporal feature learning in videos. The construction of ARTNet is based on a generic building block, termed as SMART, which aims to model appearance and relation separately and explicitly with a two-branch unit. As demonstrated on the Kinetics dataset, SMART block is able to yield better performance than the 3D convolution, and ARTNet with a single RGB input even outperforms the C3D with two-stream inputs. For representation transfer from Kinetics to datasets of UCF101 and HMDB51, ARTNet also achieves superior performance to the original C3D.

Paper 16: Beyond Short Snippets: Deep Networks for Video Classification

By: Joe Yue-Hei Ng, Matthew Hausknecht, Sudheendra Vijayanarasimhan, Oriol Vinyals, and Rajat Monga

Convolutional Neural Networks have proven highly successful at static image recognition problems such as the MNIST, CIFAR, and ImageNet Large-Scale Visual Recognition Challenge. By using a hierarchy of trainable filters and feature pooling operations, CNNs are capable of automatically learning complex features required for visual object recognition tasks achieving superior performance to hand-crafted features. Encouraged by these positive results several approaches have been proposed recently to apply CNNs to video and action classification tasks. Since each individual video frame forms only a small part of the video's story, such an approach would be using incomplete information and could therefore easily confuse classes especially if there are fine-grained distinctions or portions of the video irrelevant to the action of interest.

The contribution can be summarized as:

1. CNN architectures for obtaining global video-level descriptors and demonstrate that using increasing numbers of frames significantly improves classification performance.
2. By sharing parameters through time, the number of parameters remains constant as a function of video length in both the feature pooling and LSTM architectures.

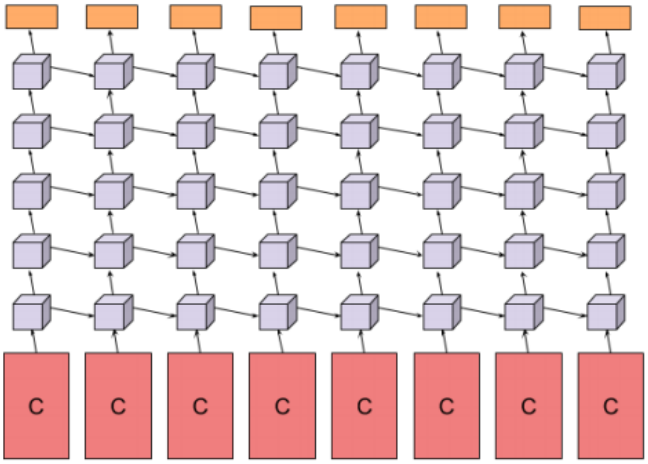
3. Optical flow images can greatly benefit video classification and present results showing that even if the optical flow images themselves are very noisy (as is the case with the Sports-1M dataset), they can still provide a benefit when coupled with LSTMs.

Amongst related works, LSTM and CNN networks had been used before. SIFT had been used in order to extract feature from frames. Long Short Term Memory (LSTM) [11] uses memory cells to store, modify, and access internal state, allowing it to better discover long-range temporal relationships. For this reason, LSTMs yield state-of-the-art results in handwriting recognition, speech recognition and emotion detection. Two CNN architectures are used to process individual video frames: AlexNet and GoogLeNet. AlexNet is a Krizhevsky-style CNN which takes a 220 × 220 sized frame as input. This frame is then processed by square convolutional layers of size 11, 9, and 5 each followed by max- pooling and local contrast normalization. Finally, outputs are fed to two fully-connected layers each with 4096 rectified linear units (ReLU). Dropout is applied to each fullyconnected layer with a ratio of 0.6 (keeping and scaling 40% of the original outputs).

GoogLeNet, uses a network-in-network approach, stacking Inception modules to form a network 22 layers deep that is substantially different from previous CNNs. Like AlexNet, GoogLeNet takes a single image of size 220 × 220 as input. This image is then passed through multiple Inception modules, each of which applies, in parallel, 1×1, 3×3, 5×5 convolution, and max-pooling operations and concatenates the resulting filters. Finally, the activations are average-pooled and output as a 1000- dimensional vector.

Typically, image based or motion features are computed at every frame, quantized, and then pooled across time. The resulting vector can be used for making video-level predictions. The pooling operation need not be limited to max pooling. Max pooling generates much sparser updates, and as a result tends to yield networks that learn faster, since the gradient update is generated by a sparse set of features from each frame. Therefore, in the rest of the paper max pooling was used as the main feature aggregation technique.

After this the architecture of different pooling layers like Conv Pooling, Late pooling, slow pooling, local pooling etcetera were explained. Up next the LSTM architecture was explained. Deep video LSTMs takes input from the CNN layers. Finally a softmax layer predicts the class at the end.



The max-pooling models were optimized on a cluster using Downpour Stochastic Gradient Descent starting with a learning rate of 10−5 in conjunction with a momentum of 0.9 and weight decay of 0.0005. we used the same optimization method with a learning rate of N ∗ 10−5 where N is number of frames. The learning rate was exponentially decayed over time. Each model had between ten and fifty replicas split across four partitions. To reduce CNN training time, the parameters of AlexNet and GoogLeNet were initialized from a pre-trained ImageNet model and then fine-tuned on Sports-1M videos.

Multi-frame models achieve higher accuracy at the cost of longer training times than single-frame models. Since pooling is performed after CNN towers that share weights, the parameters for a single-frame and multi-frame

max-pooling network are very similar. This makes it possible to expand a single-frame model to a multi-frame model. In LSTM training, all the features are trained sequentially. As the long-short term memory is concerned, the order of data coming as one after the other is very much relatable. The movement of object is a moving video frame is an important feature to record. Therefore, when that is dealt with, it becomes easy for the model to get trained on that data. The model uses the sequential data, that is the change in position of the object from every frame to the next frame.

Method	Clip Hit@1	Hit@1	Hit@5
Conv Pooling	**68.7**	**71.1**	**89.3**
Late Pooling	65.1	67.5	87.2
Slow Pooling	67.1	69.7	88.4
Local Pooling	68.1	70.4	88.9
Time-Domain Convolution	64.2	67.2	87.2

Method	Hit@1	Hit@5
AlexNet single frame	63.6	84.7
GoogLeNet single frame	**64.9**	**86.6**
LSTM + AlexNet (fc)	62.7	83.6
LSTM + GoogLeNet (fc)	**67.5**	**87.1**
Conv pooling + AlexNet	70.4	89.0
Conv pooling + GoogLeNet	**71.7**	**90.4**

Method	Frames	Clip Hit@1	Hit@1	Hit@5
LSTM	30	N/A	72.1	90.4
Conv pooling	30	66.0	71.7	90.4
	120	70.8	72.3	90.8

Table 3: Effect of the number of frames in the model. Both LSTM and Conv-Pooling models use GoogLeNet CNN.

Method	Hit@1	Hit@5
LSTM on Optical Flow	59.7	81.4
LSTM on Raw Frames	72.1	90.6
LSTM on Raw Frames + LSTM on Optical Flow	73.1	90.5
30 frame Optical Flow	44.5	70.4
Conv Pooling on Raw Frames	71.7	90.4
Conv Pooling on Raw Frames + Conv Pooling on Optical Flow	71.8	90.4

The accuracies based upon Frame rates are as such:

Method	Frame Rate	3-fold Accuracy (%)
Single Frame Model	N/A	73.3
Conv Pooling (30 frames)	30 fps	80.8
	6 fps	82.0
Conv Pooling (120 frames)	30 fps	82.6
	6 fps	82.6

The final accuracies on the UCF101 dataset is:

Method	3-fold Accuracy (%)
Improved Dense Trajectories (IDTF)s [23]	87.9
Slow Fusion CNN [14]	65.4
Single Frame CNN Model (Images) [19]	73.0
Single Frame CNN Model (Optical Flow) [19]	73.9
Two-Stream CNN (Optical Flow + Image Frames, Averaging) [19]	86.9
Two-Stream CNN (Optical Flow + Image Frames, SVM Fusion) [19]	88.0
Our Single Frame Model	73.3
Conv Pooling of Image Frames + Optical Flow (30 Frames)	87.6
Conv Pooling of Image Frames + Optical Flow (120 Frames)	**88.2**
LSTM with 30 Frame Unroll (Optical Flow + Image Frames)	**88.6**

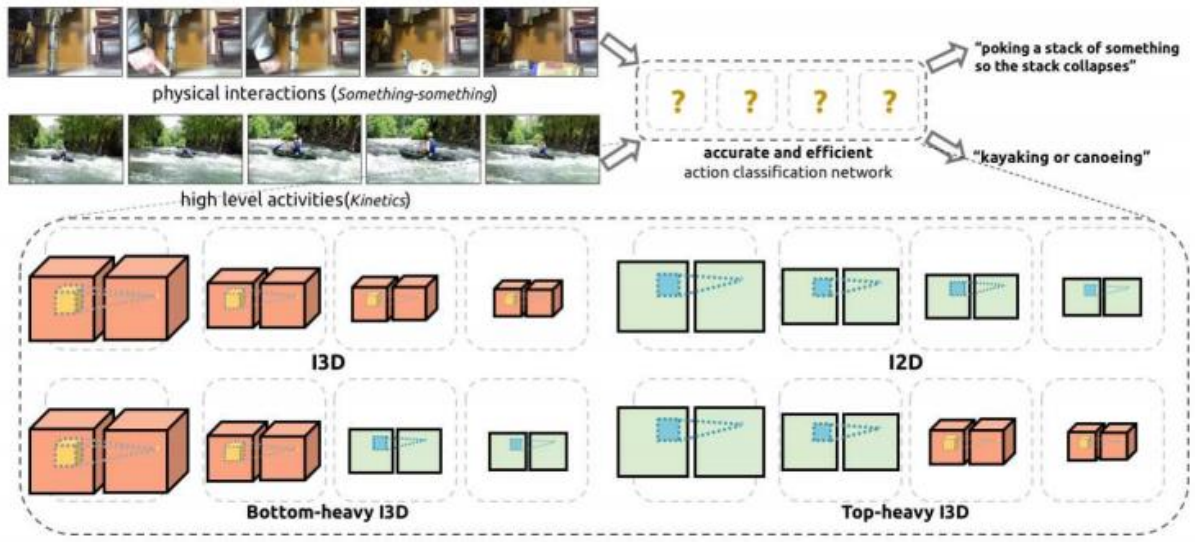
In the current models, backpropagation of gradients proceeds down all layers and backwards through time in the top layers, but not backwards through time in the lower (CNN) layers. In the future, it would be interesting to consider a deeper integration of the temporal sequence information into the CNNs themselves. For instance, a Recurrent Convolutional Neural Network may be able to generate better features by utilizing its own activations in the last frame in conjunction with the image from the current frame.

Paper 17: Rethinking Spatiotemporal Feature Learning: Speed-Accuracy Trade-offs in Video Classification

By: Saining Xie1, Chen Sun, Jonathan Huang, Zhuowen Tu, and Kevin Murphy

Google Research 2 University of California, San Diego

The resurgence of convolutional neural networks (CNNs) has led to a wave of unprecedented advances for image classification using end-to-end hierarchical feature learning architectures. The task of video classification, however, has not enjoyed the same level of performance jump as in image classification.



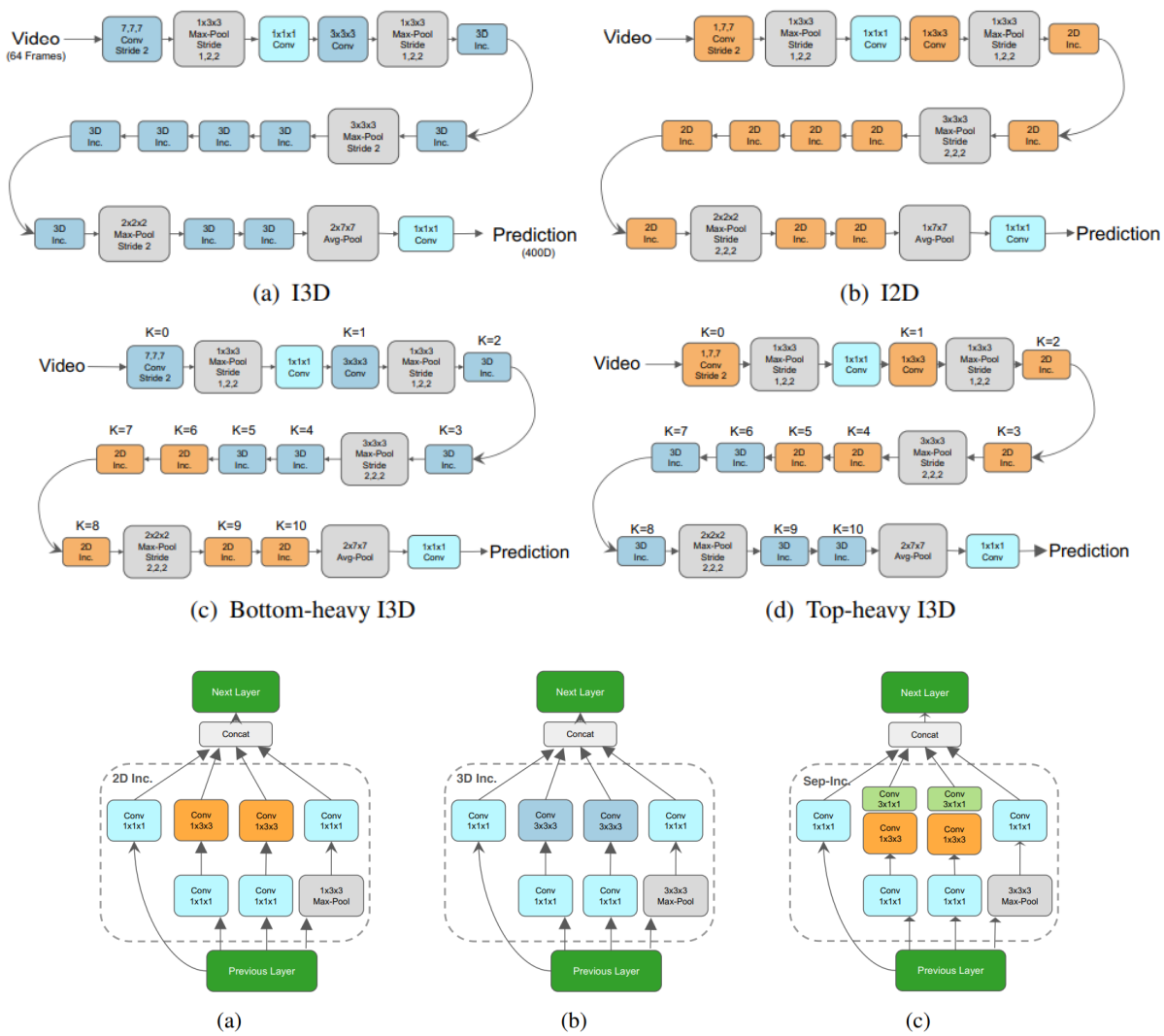
2D CNNs have achieved state of the art results for image classification, so, not surprisingly, there have been many recent attempts to extend these successes to video classification. There are three key ingredients for its success: first, they “inflate” all the 2D convolution filters used by the Inception V1 architecture into 3D convolutions, and carefully choose the temporal kernel size in the earlier layers. Second, they initialize the

inflated model weights by duplicating weights that were pre-trained on ImageNet classification over the temporal dimension. Finally, they train the network on the large-scale Kinetics dataset.

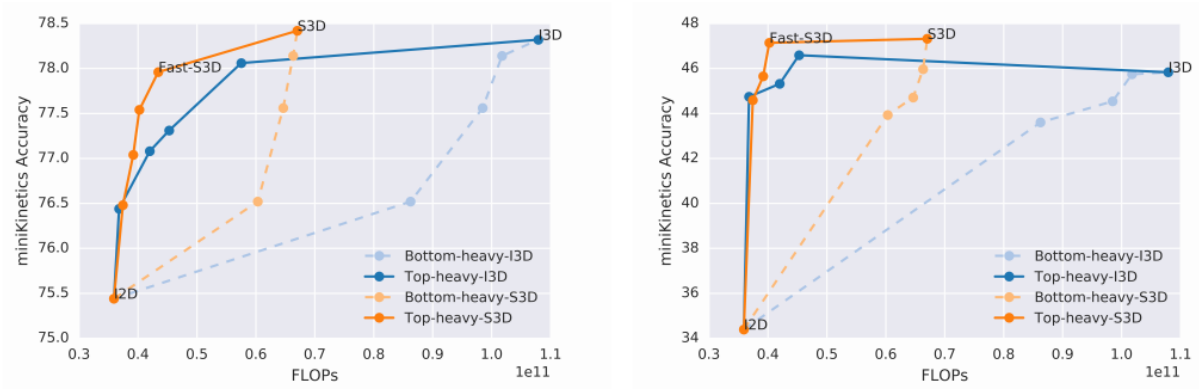
In this paper, we consider two large video action classification datasets. The first one is Kinetics [6], which is a large dataset collected from YouTube, containing 400 action classes and 240K training examples. Each example is temporally trimmed to be around 10 seconds. Since the full Kinetics dataset is quite large, we have created a smaller dataset that we call Mini-Kinetics-200.3 Mini-Kinetics-200 consists of the 200 categories with most training examples; for each category, we randomly sample 400 examples from the training set, and 25 examples from the validation set, resulting in 80K training examples and 5K validation examples in total. The splits are publicly released to enable future comparisons. We also report some results on the original Kinetics dataset, which we will call Kinetics-Full for clarity.

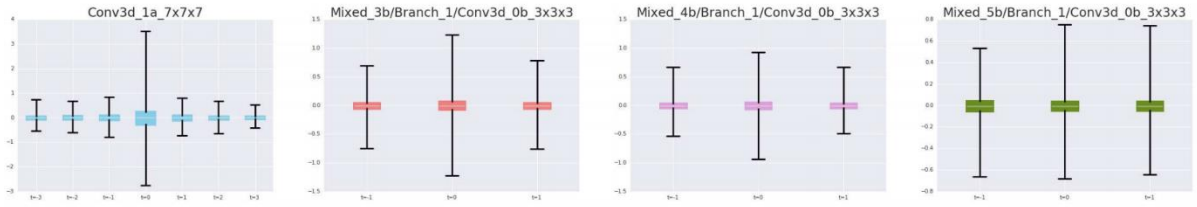
During training, we densely sample 64 frames from a video and resize input frames to 256 × 256 and then take random crops of size 224×224. During evaluation, we use all frames and take 224×224 centre crops from the resized frames. Our models are implemented with Tensorflow and optimized with a vanilla synchronous SGD algorithm with momentum of 0.9 and on 56 GPUs, batch size is set to 6 per GPU.

Replacing 3D convolutions with 2D convolutions,

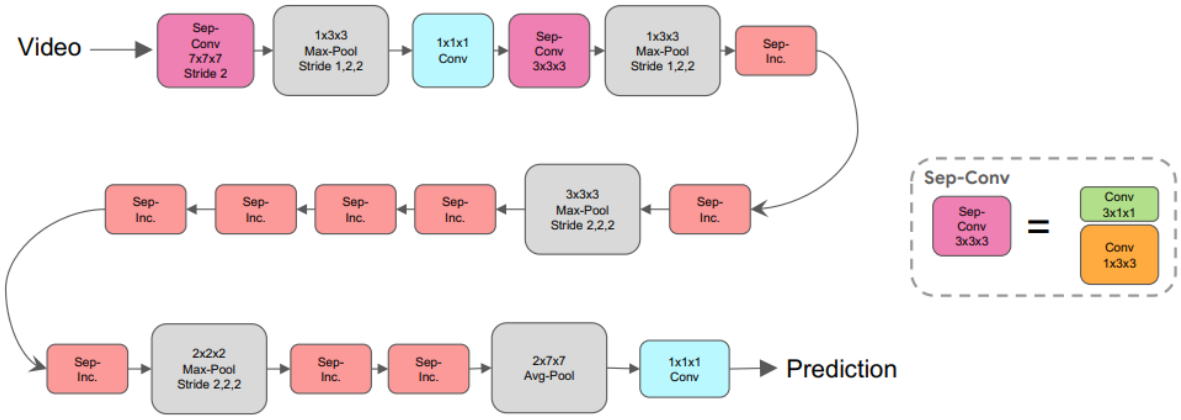


Analysis of weight distribution of learned filters:

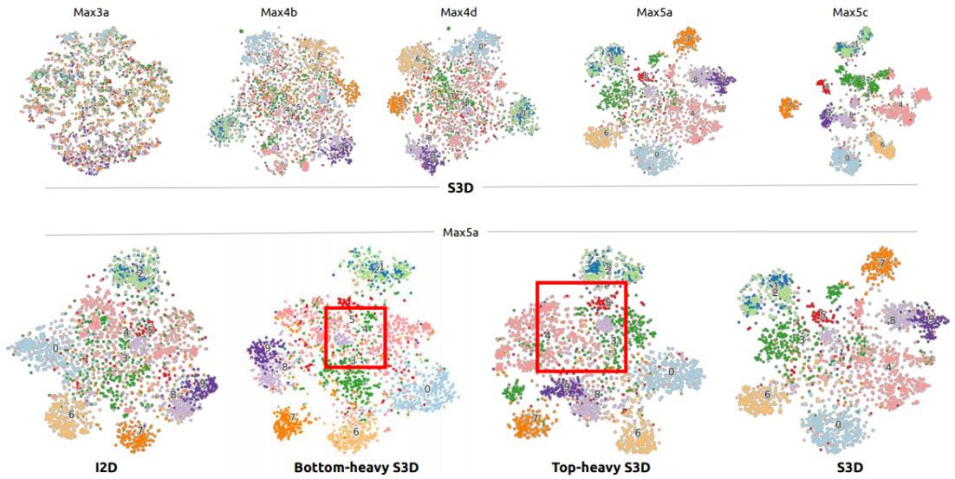




An architecture of S3D model used:



Colours and numbers depict ten classes. TSNE based projections are shown:



Up next TSNE is explained and it's features were deduced to train and test. Then after optical flow was considered in order to figure out the motion of object between frames. Fine-tuning of the datasets are done and that tuning is done per frame.

Finally, we demonstrate the effectiveness of S3D-G on action detection tasks, where the inputs are video frames, and the outputs are bounding boxes associated with action labels on the frames. Similar to the framework proposed in, we use the FasterRCNN object detection algorithm to jointly perform person localization and action recognition. We use the same approach as described in to incorporate temporal context information via 3D networks. To be more specific, the model uses a 2D ResNet50 network that takes the annotated key frame (frame with box annotations) as input, and extract features for region proposal generation on the key frame.

The results are as follows:

Model	Inputs	JHMDB	UCF-101
Gkioxari and Malik [54]	RGB+Flow	36.2	-
Weinzaepfel *et al.* [56]	RGB+Flow	45.8	35.8
Peng and Schmid [49]	RGB+Flow	58.5	65.7
Kalogeiton *et al.* [57]	RGB+Flow	65.7	69.5
Faster RCNN + I3D [51]	RGB+Flow	73.2	76.3
Faster RCNN + S3D-G	RGB+Flow	**75.2**	**78.8**

Improving on the previous state of the art 3D CNN video classification model, known as I3D, in terms of efficiency, by combining 3 key ideas: a top-heavy model design, temporally separable convolution, and Spatio-temporal feature gating. Our modifications are simple and can be applied to other architectures. We hope this will boost performance on a variety of video understanding tasks.

Paper 18: Multimodal Keyless Attention Fusion for Video Classification

By: Xiang Long, Chuang Gan , Gerard de Melo, Xiao Liu, Yandong Li, Fu Li, Shilei Wen

The Thirty-Second AAAI Conference on Artificial Intelligence (AAAI-18)

Unlike image classification, which takes still pictures as input, video classification is inherently multimodal, and thus image, motion, as well as sound cues may be necessary to make a comprehensive judgment. For instance, two musical instruments may be difficult to distinguish based on their mere appearance in a video, but might produce rather distinct sounds. In this case, acoustic features can greatly improve the accuracy of a video classification model. As in other areas of computer vision, approaches based on deep convolutional neural networks (CNNs) have achieved state-of-the-art results. However, the improvements brought by CNNs, given their focus on local patterns, have not been as pronounced as for images. Due to the temporal sequential nature of videos, which can be very long, recurrent networks (RNNs) may be invoked to better capture longer-range temporal patterns and relationships. However, existing end-to-end approaches are restricted to small-scale datasets. It remains very difficult to combine CNN and RNN modelling for joint end-to-end training directly on large-scale datasets such as Kinetics and YouTube-8M. Overall summary of the paper is to:

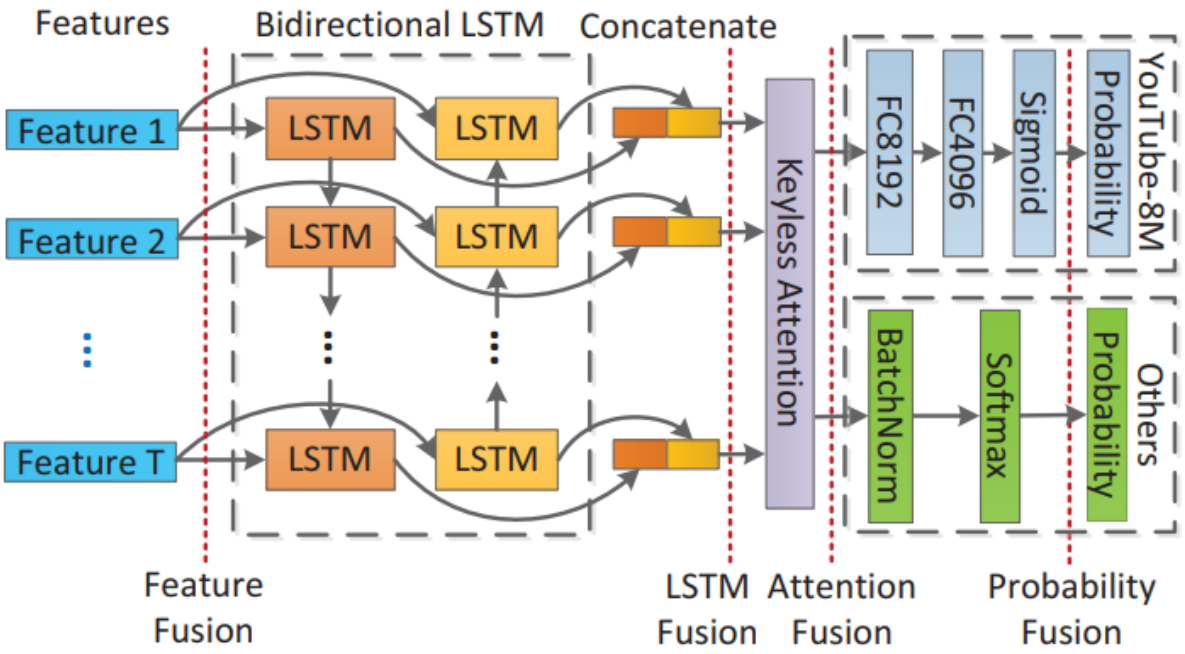
1. Propose and implement simple RNN model.
2. Analysis of various fusion-based methods of RNN based architectures.
3. Highly competitive results on both the standard UCF-101 and ActivityNet datasets are obtained.

Humans recognize objects and events not by processing an entire visual scene simultaneously, but by selectively focusing on parts of the scene that provide the most pertinent information. Attention models were first proposed for object recognition with recurrent neural networks, drawing on the

REINFORCE algorithm. natural language processing and visual captioning. In particular, Bahdanau et al. aimed at automatically capturing soft alignments between source words and target words in machine translation.

It stacks another RNN for motion modelling to better guide the attention towards the relevant spatial-temporal regions. However, these complex attention architectures are highly integrated with RNNs, and need to recalculate the attention weight map at every iteration. The attention modeling thus adds a significant burden to the computation and does not bring sufficient improvements in accuracy.

Video is an inherently multimodal medium, with both visual and sound modalities. The video signal, moreover, can further be decomposed into static frames on the one hand, and signals capturing continuous motion on the other. Hence, using features of a single modality is clearly inadequate. In this paper, we use the following multimodal features to more thoroughly represent the contents of a video.



A deep convolutional networks is meant to extract acoustic features by pre-processing the raw audio to emit a sequence of matrices. The audio is first divided into non-overlapping 960ms frames. The frames are then decomposed with a short-time Fourier transform every 10ms and then aggregated, logarithm-transformed, into 64 Mel-spaced frequency bins following.

Despite having obtained frame-level visual and acoustic features, we do not simply feed these into a recurrent network. First, the number of frames in a video can range to several thousands, which makes a direct application of LSTMs very challenging, as even LSTMs often fail to capture particularly long-range dependencies.

Afterwards, keyless attention layer is defined through a set of equations. For using multiple layers of features, multimodal fusion is used. However, in order to fully exploit the multimodal nature of videos, one needs to account for multiple modalities.

A simple feature-level fusion is one of the most intuitive methods. Each feature represents a particular temporal segment in the video. Stitching together the features of the same segment leads to a more detailed representation of that temporal segment. Further on Attention Fusion and Probability Fusion is defined. For datasets, UCF101, YouTube-8M, Kinetics and ActivityNet are used.

As for results, the results on 4 datasets are as follows:

Method		Accuracy(%)
iDT + FV (Wang and Schmid 2013)		85.9
iDT + HSV (Peng et al. 2016)		87.9
EMV-CNN (Zhang et al. 2016)		86.4
Two Stream (Simonyan and Zisserman 2014b)		88.0
FSTCN (Sun et al. 2015)		88.1
VideoLSTM(Li et al. 2016)		89.2
TDD+FV (Wang, Qiao, and Tang 2015)		90.3
Fusion (Feichtenhofer, Pinz, and Zisserman 2016)		92.5
TSN(3 seg) (Wang et al. 2016a)		94.2
ST-ResNet+iDT (Feichtenhofer, Pinz, and Wildes 2016)		94.6
ActionVLAD (Girdhar et al. 2017)		93.6
Ours	RGB CNN	85.4
	RGB Average	85.9
	RGB Last	85.8
	RGB Attention	**86.2**
	Flow CNN	86.2
	Flow Attention	**87.0**
	Feature Fusion	94.1
	LSTM Fusion	94.2
	Probility Fusion	93.5
	Attention Fusion	**94.8**

Table 1: Mean classification accuracy on UCF-101.

Method		mAP(%)
iDT + FV (Peng et al. 2016)		66.5
Depth2Action (Zhu and Newsam 2016)		78.1
Two Stream (Simonyan and Zisserman 2014b)		71.9
C3D (Tran et al. 2015)		74.1
Ours	RGB CNN	70.9
	RGB Average	71.3
	RGB Last	71.4
	RGB Attention	**72.0**
	Flow CNN	64.0
	Flow Attention	**64.8**
	Feature Fusion	77.5
	LSTM Fusion	77.6
	Probility Fusion	77.2
	Attention Fusion	**78.5**

Table 2: ActivityNet results on the validation set.

Results for Multimodal attention:

Method		Top-1(%)	Top-5(%)
C3D (Tran et al. 2015)		55.6	79.1
3D ResNet (Hara, Kataoka, and Satoh 2017)		58.0	81.3
Two-Stream I3D* (Carreira and Zisserman 2017)		74.2	91.3
Ours	RGB CNN	73.0	90.9
	RGB Average	73.2	91.1
	RGB Last	73.0	91.0
	RGB Attention	**73.8**	**91.3**
	Flow CNN	54.5	75.9
	Flow Attention	**54.9**	**76.4**
	Audio CNN	21.6	39.4
	Audio Attention	**22.0**	**40.1**
	Feature Fusion	76.1	92.6
	LSTM Fusion	76.2	92.6
	Probility Fusion	74.9	91.6
	Attention Fusion	**77.0**	**93.2**

Table 3: Kinetics results on the validation set, except for those marked with '∗', which are based on the test set.

Method		60K Valid(%)	Test(%)
VLAD (Xu, Yang, and Hauptmann 2015)		-	80.4
Video Level (Zhong et al. 2017)		-	78.6
LSTM + MoE (Wang et al. 2017)		-	80.2
Ours	RGB Attention	76.6	77.3
	Audio Attention	54.0	-
	Feature Fusion	80.5	81.5
	Probility Fusion	79.1	-
	Attention Fusion	**80.9**	**82.2**

Table 4: YouTube-8M GAP@20 on the 60K validation and test set.

To better cope with the sequential and multimodal nature of videos, we have proposed a keyless attention mechanism, which allows for fast and effective learning of RNN models. We further assess several alternatives for achieving multimodal fusion with recurrent neural networks, and find that the proposed attention-based fusion achieves the best results. We have conducted experiments on four well known datasets, including both untrimmed and trimmed videos, single-label and multi-label classification settings, and small-scale as well as very large-scale datasets. Our highly competitive

results in all of these settings demonstrate that our proposed Multimodal Keyless Attention Fusion framework is robust across a large range of video classification tasks.

Paper 19: Crowd Video Event Classification using Convolutional Neural Network

By: S. Jothi Shri a, S. Jothilakshmi

Computer Communications 147 (2019) 35–39
Received 11 June 2019; Received in revised form 10 July 2019; Accepted 30 July 2019

Crowd Event Classification in videos has great potential in many applications and it is useful to detect pertained crowd events from the input videos and classify them. An event is a recognizable occurrence in scenes of videos and is characterized by the subject. The Crowd Event Classification is a difficult process due to various subject contained in the input videos. The subject of the event can be human action recognition, motions, and places. Deep Convolutional Neural Networks have significantly advanced in computer vision and natural language processing. The CNN network is operative in the performance of describing the high level and difficult features of video input abstraction data over a hierarchical learning method. Deep Neural Network achieves superior performance in visual object recognition. The large scale events are taken as the classes for evaluating statistics of features. The explorations of objects are identified and tracked their pose from the events. The process of detecting entities is a difficult and challenging task. The main focus of the event classification has been on techniques based only by using statistical features. , VGG-16 network model is used to describe the video event features with 13 Convolutional, 5 Max pooling and 3 Fully connected layers with another Soft-Max layer. The system is trained by ImageNet network model with 1000 categories to describe the video event features. The classifier learned to classify the extracted features.

The video contains visual information of the event can be processed on a frame basis using Convolution Neural Network (CNN). The system event has randomly initialized Deep CNN and performed on high-resolution video frames. The huge amount of labelled statistics data used for working out the model. The classifier learns the relative significance of the features from the training set of data. The CNN introduces two approaches namely baseline and VGG-16 feature representations to improve the effectiveness of the model. The model introduces temporal (video frames) connectivity of architecture to improve the computational time. The significant improvements are identified from the results of two different representations of the CNN model.

Convolutional layers, pooling layers, and fully connected layers are contained in the CNN architecture as shown. The convolutional layers are performing in the main function of CNN. The convolutional layer is connected nearby each neuron to a small region of the video input data at the same time. The region is called a receptive field. Each neuron of CNN layers is considered by a dot product among its weights and the small input region. The neurons are described by the whole area of the image and it is covered by the respective fields of every neuron. Every neuron performed by sliding the layer weights. Both The resulting convolution output and the input image are the same size ($224 \times 224 \times 3$). The receptive field depth gets all the colour information from the input image. The receptive field is the same as the input colour channel.

The CNN architecture is shown:

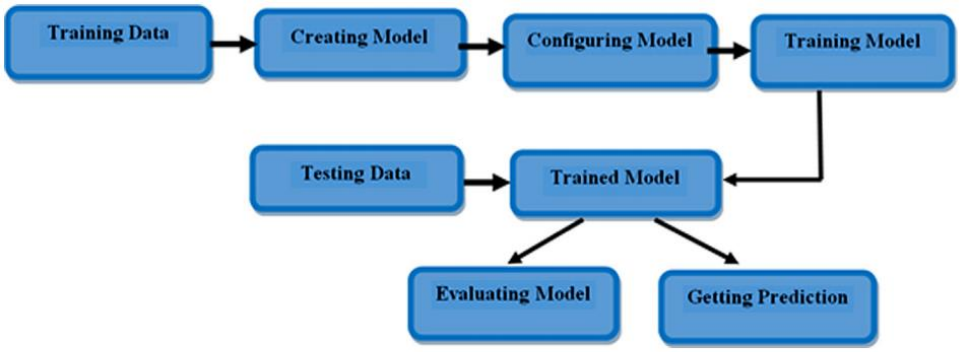
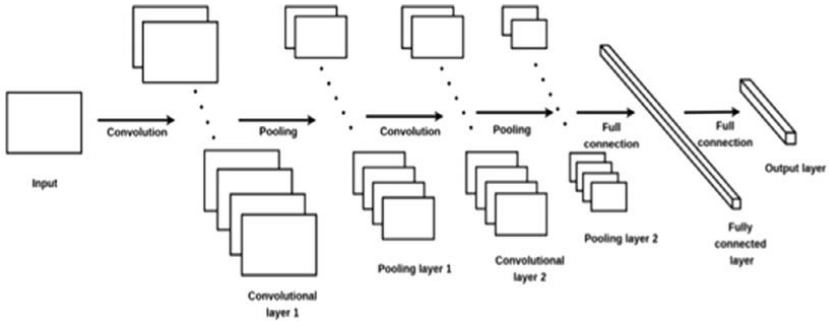
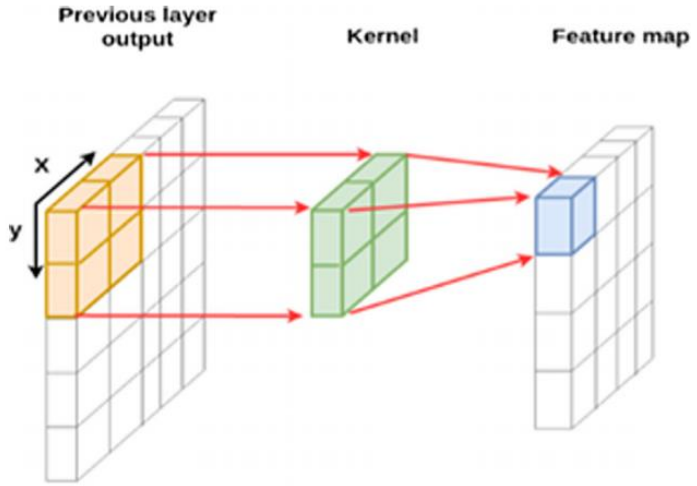


Fig. 1. Steps involved in the proposed system.



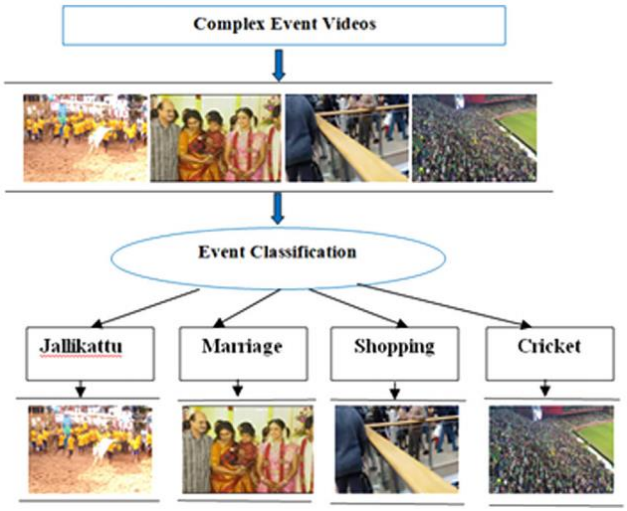
The architecture of Kernel of network layer:



The proposed system implements the VGG16 model to compare the performance with the baseline model. In VGG-16 model has the input to the convolution layer is 150 × 150 image in a fixed size. The picture is carried through a stack of convolutionary layers where the filters were used with a very tiny receptive field 3 × 3, the smallest size to capture the concept of left, right, up, down, middle. It also uses 1 × 1 random convolution filters in one of the configurations, which can be viewed as a linear input channel transformation. The convolution step is set at 1 pixel; convolution spatial padding. The input of the layer is such that after convolution the spatial resolution is maintained, i.e. the padding is 1-pixel for 3 × 3 sequence layers. The convolution step is set at 1 pixel; Conv spatial padding. Spatial pooling is performed by five layers of max pooling that follow some of the convolution layers. Max-pooling takes place over a window of 22 pixels, with step 2. All concealed layers are fitted with non-linearity of rectification (ReLU). It is also observed that none of the networks (except one) contain

Local Response Standardization (LRN), such standardization does not enhance the efficiency of the ILSVRC dataset, but results in enhanced memory consumption and computation time.

The model studied the performance of frame-based video recognition using features from the layers of a deep convolutional model organized with various kernels for event classification.



The proposed system applies Convolutional Neural Network (CNN) to classify 4000 data samples and reports major developments in the result. The results of baselines and VGG-16 gives the result based on Features established by training networks. A Deep Learning is established on event classifier trained through 3000 frames of videos. First, randomly selected 1000 images per event category are a training set and 1000 images are a validation set for 4 categories. The deep CNN achieved 100% event classification accuracy on the validation set after training. Deep learning algorithm has been implemented by Python 3.5 version along with Anaconda Library.

Table 1
Computation result of the experiment.

Epoch	Baseline training accuracy	VGG 16 training accuracy
1	73	82
2	100	90
3	100	96
4	100	82
5	99	88
6	100	92

Table 2
Performance of crowd event classification system.

Model	No. of dataset			
	True positive	True negative	False positive	False negative
CNN baseline	100	40	0	0
VGG16	100	40	40	8

The following are the results:

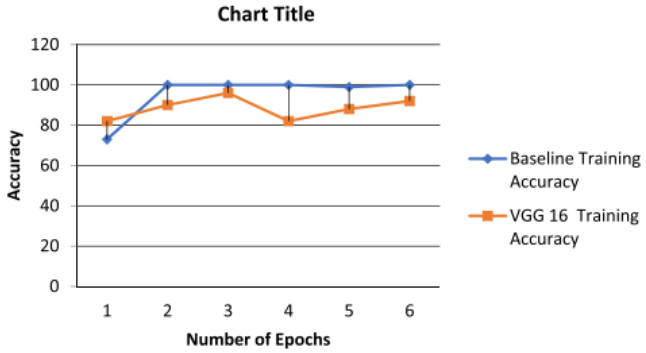
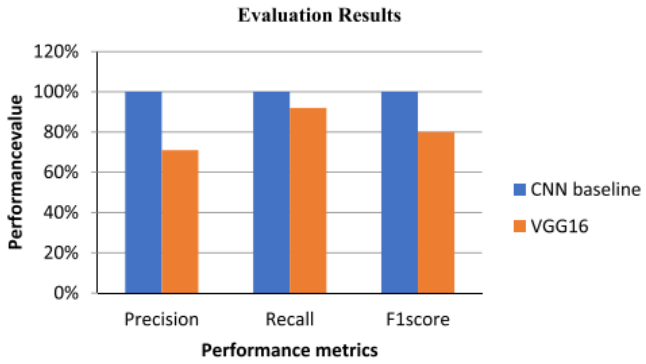


Fig. 5. CNN evaluation of accuracy on training.

Table 3
Computation results of the experiment.

Model	Precision	Recall	F1score
CNN baseline	100%	100%	100%
VGG16	71%	92%	80%



The Crowd event classification system provides a Convolutional Neural Network model to classify the crowd events collected from YouTube videos. The proposed system trains the model of CNN with 4000 video frames of four categories. There is two feature representation of approaches are implemented namely baseline and VGG16 model to classify the video frames. The performance of the two approaches is compared and showed. The VGG16 model gives the 82% result and it shows inconstant accuracy in continuous epochs. The proposed system has found the best result of 100% constant in continuous epochs in this work suggests a baseline approach for crowd event classification. The baseline model shows improved runtime performance at low cost. In future work, the high number of crowd video event dataset will be considered to classify video events in the best way. Mainly the proposed may use this work in video surveillance applications.

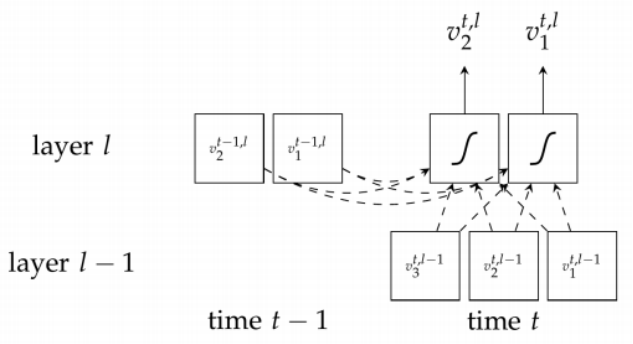
Paper 20: Recurrent Convolutional Neural Network for Video classification

By: Zhenqi Xu, Jiani Hu, and Weihong Deng

Beijing University of Posts and Telecommunications

Deep learning has been used broadly in many machine learning area due to its great performance recently. A good strategy to apply deep learning architecture in a task is to incorporate the prior knowledge of that task in the architecture. For example, convolutional neural network (CNN) for image tasks can utilize the image prior: pixels that are spatially nearby are highly correlated [5]. This local structure is extended to time dimension in videos, that is, the pixels in the same spatial position between consequent frames are also highly correlated. Pixels in different frames form motion (or temporal) features. This correlation induces a lot of redundancy in videos. Failure to deal with this redundancy may cause much computation and poor performance, which makes the video

classification using deep learning methods difficult. Existing deep learning methods always recur to optical flow computed from consequent image frames to incorporate motion features [12]. The two-stream features are complementary to each other, thus fusion of optical flow and raw image frames on the score level can boost performance. This shows two things, firstly, the motion features are actually important for action recognition. Secondly, the existing architecture can not model the motion feature from raw images well.

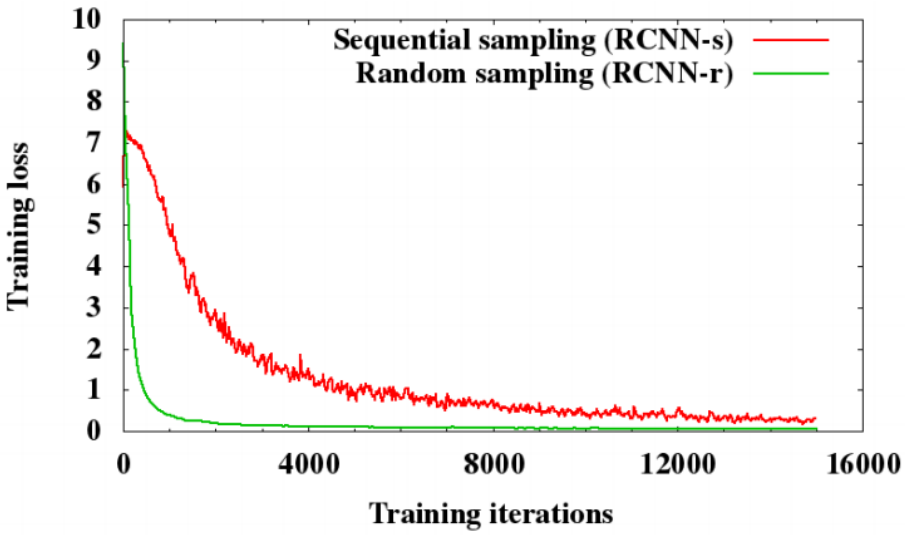


The goal here is to fully use the prior knowledge of videos in designing our deep learning architecture. A good architecture for video classification should 1) utilize the local structure in image frames, 2) incorporate the motion features between image frames 3) reduce the redundancy of videos. We deal with video classification problem by extending traditional RNN to recurrent convolutional neural network, which can learn local and dense feature through convolution operation as well as modelling motion between image frames by the linking of previous time step. Lastly, we can visualize the motion features just like the spatial features, which is helpful for understanding the learned models and diagnosing the model when training. We achieve 81.0% accuracy on UCF101 dataset without using optical flow and 86.3% with optical flow.

In traditional RNNs, each layer has a connection from outputs of the previous time step. The layer firstly do a linear transformation for both inputs from the previous time step and inputs from the present layer. Then sum of these two parts are through a non-linear function, usually sigmoid function, tanh function or recently rectified linear unit (ReLU). However, the traditional RNN may not work well for videos, since it doesn't incorporate the prior knowledge that images are locally relevant. CNN can model this prior knowledge, but ignores the motion between image frames.

We have done several experiments on the UCF-101 dataset which contains 101 action classes with total 13320 videos. We follow the standard evaluation protocol, use the first split to compare models and report the average classification accuracy over three splits. When training the GoogleNet-RCNN models, two additional signals are added to the network to gain fast learning, we denote the three

signals as low signal, medium signal, high signal, and test their performance. The higher signal is better than medium signal and outperform the low signal by a large margin. This means that the good property of deep structure also benefit our RCNN layer.



The overall performance results is:

Method	3-fold Accuracy		
	image	optical flow	two-stream
Improved dense trajectories (IDT) [3]	85.9%		
IDT with encodings [4]	87.9%		
Slow fusion CNN [7][1]	65.4%	–	–
Conv pooling CNN [6][1]	**82.6%**	–	88.2%
CNN + LSTM [6][1]	–	–	**88.6%**
CNN + SVM [12][2]	73.0%	83.7%	88.0%
CNN + LSTM [10]	71.1%	77.0%	82.9%
RCNN-s (ours)	74.1%	–	–
RCNN-r (ours)	**81.0%**	76.4%	86.3%

The network can extract spatial feature by convolution operation and learning the temporal feature by recurrent linking of the neuron layers. Thus, it is appropriate for video classification tasks. Besides, we explore random sampling and sequential sampling when deal with videos, and prove that random sampling is necessary for video classification due to the large redundancy in videos.