Literature Review: Action Recognition

Author: Shivam Sharma

Abstract

Deep learning approaches have empirically demonstrated remarkable success in learning image representations for tasks like object recognition, image captioning, and semantic segmentation. Convolutional neural networks have enabled us to efficiently capture the hypothesis of spatial locality of data structure in images through parameter sharing convolutions and local invariance-building neurons called max-pooling. In this literature review, we would like to explore the impact of deep learning techniques on video tasks, specifically action recognition. We would like to explore how spatiotemporal features are aggregated through various deep architectures, the role of optical flow as an input, the impacts on real-time capabilities, and the compactness & interpretability of the learned features. We will then propose areas of future research that we believe could help bias our deep learning architectures in a way that captures temporal hypotheses of the real-world.

Introduction

Action recognition has important applications in smart surveillance, human-computer interaction, and video search. As much of the content we consume and produce comes in the form of video, it is increasingly important for us to understand how we can correctly discern temporal features from spatial features. This will help us create learners in the future that can make predictions or decisions from a richer understanding of the 3D environment we live in.

Survey

Before Deep Learning

Before 2014, the state of the art techniques focused on hand crafted features formed from sparsely or densely sampled trajectories. For example, a popular approach called Improved Dense Trajectories (iDT) [1], extracted trajectories and features for a dense set of interest points, encodes them in a fixed sized video description, then a classifier like SVM is trained on the resulting "bag of words" representation. In this approach, there is a lot of preprocessing that needs to be done for each frame. The optical flow must be calculated between frames, gradients for optical flow are also calculated, and mean-subtracted histograms are produced for both. These are all considered input features that are encoded into the fixed size video description. We will see in deep learning approaches to video representation how preprocessing will have an effect on being end-to-end trainable as well as on real-time capabilities.

Post Deep Learning

After 2014, deep learning architectures prevailed with state of the art performance on landmark video action recognition datasets like UCF101, Sports-1M, and HMDB51. In 2014, two important breakthrough papers gave deep learning the start in video recognition. *Large-scale Video Classification with Convolutional Neural Networks* by Karpathy et. al. [2] and *Two-Stream Convolutional Networks for Action Recognition in Videos* by Simonyan and Zisserman [3] gave rise to the popularity of single stream and two stream networks in action recognition. Karpathy et. al. explored how to fuse temporal data with a single stream 2D convolutional neural net. They tested the architectures proposed in Figure 1. Their results are summarized in Table 1.

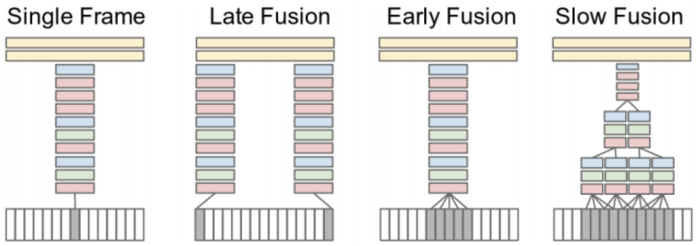


Figure 1 - Red, green and blue boxes indicate convolutional, normalization and pooling layers respectively. In the Slow Fusion model, the depicted columns share parameters. [2]

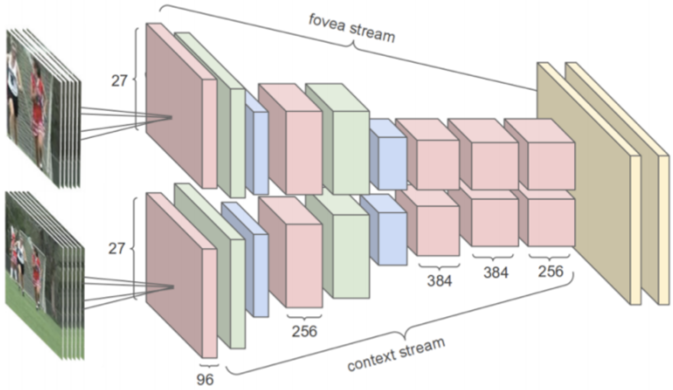
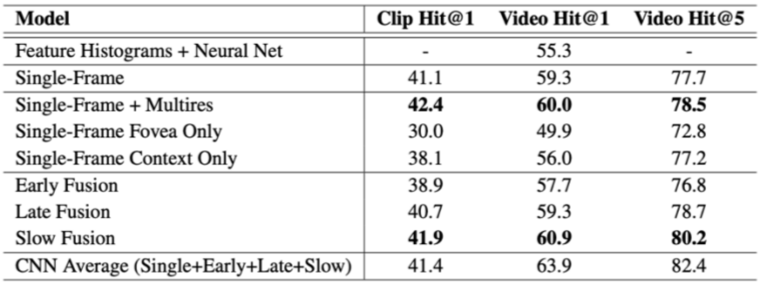


Figure 2 - Both streams consist of alternating convolution (red), normalization (green) and pooling (blue) layers. Both streams converge to two fully connected layers (yellow). [2]



Model	Clip Hit@1	Video Hit@1	Video Hit@5
Feature Histograms + Neural Net	-	55.3	-
Single-Frame	41.1	59.3	77.7
Single-Frame + Multires	**42.4**	**60.0**	**78.5**
Single-Frame Fovea Only	30.0	49.9	72.8
Single-Frame Context Only	38.1	56.0	77.2
Early Fusion	38.9	57.7	76.8
Late Fusion	40.7	59.3	78.7
Slow Fusion	**41.9**	**60.9**	**80.2**
CNN Average (Single+Early+Late+Slow)	41.4	63.9	82.4

Table 1 - Results on the 200,000 videos of the Sports-1M test set. Hit@k values indicate the fraction of test samples that contained at least one of the ground truth labels in the top k predictions. [2]

The strengths of the single stream strategy involve the fact that we can use transfer learning from models trained on large scale image datasets. We also have no need to pre-process the images for optical flow as we directly use the RGB image data in this architecture. This makes single stream networks a candidate for real-time processing. In the fusion architectures that the authors propose, the number of parameters increase significantly from a deep 2D CNN. To mitigate this, the authors propose using a multi-resolution stream. One that embeds a high-resolution fovea stream on a center crop of the video combined with a low-resolution context stream from the whole video, as seen in Figure 2. Empirically, this drastically reduced the number of parameters to an order of magnitude comparable to the single frame architecture. From the results published in Table 1, we see that the architectures proposed failed to effectively boost the performance from just using a single frame. The weakness of these models is that they did not capture motion features well. However, this paper did reveal that transfer learning is very useful for action recognition. Models that were pre-trained on Sports-1M and then finely tuned on the top 3 layers boosted accuracy on the UCF101 dataset by over 20% when compared to a model trained from scratch on UCF101 [2].

In 2014, Simonyan and Zisserman proposed a two stream architecture that processes spatial features and temporal feature separately [3] as in Figure 3. A single frame for the video is passed to a 2D convolution net while preprocessed multi-frame optical flow is passed to a separate 2D convolution net. Each stream forms a prediction and the class score is determined by their fusion. The drawback of this architecture is that it is not end-to-end trainable as optical flow needs to be calculated separately and both streams need to be trained separately. The spatial stream can take learnings from large image datasets, whereas the temporal stream must be trained on a video dataset. In this way, transfer learning is not completely applicable for this architecture. Furthermore, the preprocessing required for computing optical flow makes it difficult for this algorithm to have real-time capabilities. The strength of this approach is found in its ability to match the state of the art techniques of the time like IDT, as seen in Table 2. This opened the door for further research into deep learning for video classification. This research showed that convolutional deep nets could effectively capture some motion features and combine that with spatial features to form accurate predictions for action classes.

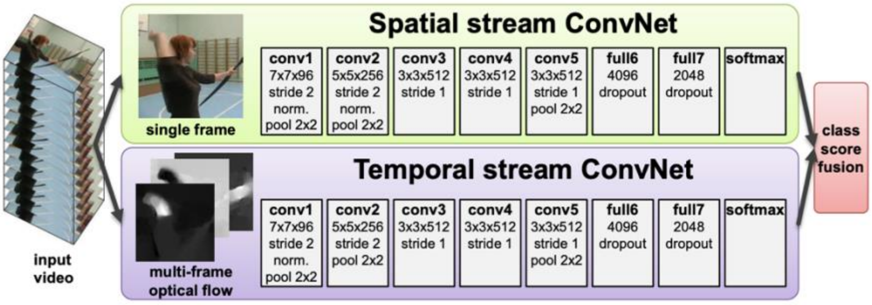
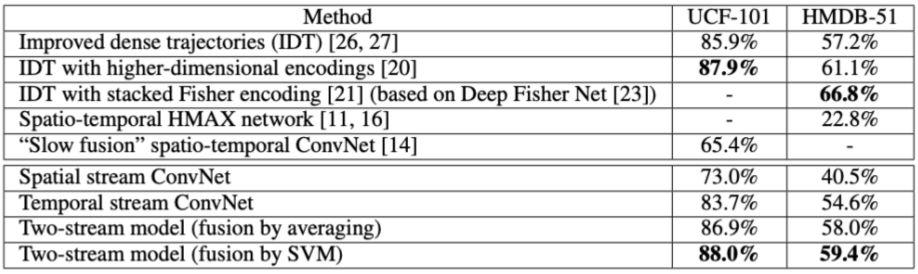


Figure 3- Individual ConvNets accuracy on UCF-101 (split 1) [3]



Method	UCF-101	HMDB-51
Improved dense trajectories (IDT) [26, 27]	85.9%	57.2%
IDT with higher-dimensional encodings [20]	**87.9%**	61.1%
IDT with stacked Fisher encoding [21] (based on Deep Fisher Net [23])	-	**66.8%**
Spatio-temporal HMAX network [11, 16]	-	22.8%
"Slow fusion" spatio-temporal ConvNet [14]	65.4%	-
Spatial stream ConvNet	73.0%	40.5%
Temporal stream ConvNet	83.7%	54.6%
Two-stream model (fusion by averaging)	86.9%	58.0%
Two-stream model (fusion by SVM)	**88.0%**	**59.4%**

Table 2 - Individual ConvNets accuracy on UCF-101 (split 1). [3]

The deep learning architectures developed in the next 5 years from 2014 to 2019 largely follow variations around the architectures depicted in Figure 4.

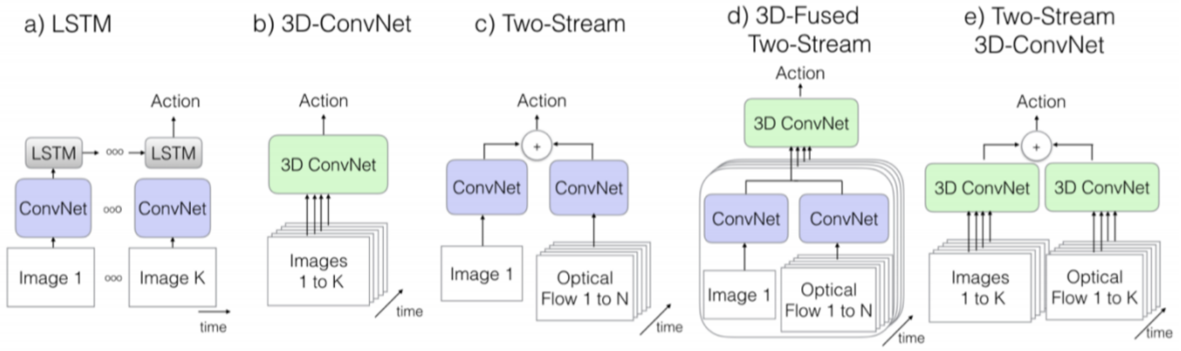
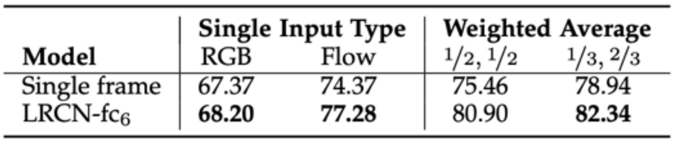


Figure 4 - Video architectures considered in this paper. K stands for the total number of frames in a video, whereas N stands for a subset of neighbouring frames of the video. [4]

The first two approaches using an LSTM and a 3D ConvNet share the strengths of being end-to-end trainable and real-time capable. This is because they do not rely on optical flow and instead must learn features that encode this information. This allows for the network to learn spatiotemporal features directly in end-to-end training. Approaches c) – e) are not real-time capable nor end-to-end trainable because they require optical flow calculations over the raw data. Approaches b), d), and e) use 3D convolutions. This creates a magnitude of more parameters from traditional 2D ConvNets. For a single 3D convolution neural network trained for the UCF101 dataset can have 33M + parameters, compared to just 5M+ parameters in the 2D case [4]. This significantly affects the training cost as 3D ConvNet models trained on Sports-1M take approximately 2 months. This makes it difficult to search for the right architecture for video data. This also creates a risk of overfitting.

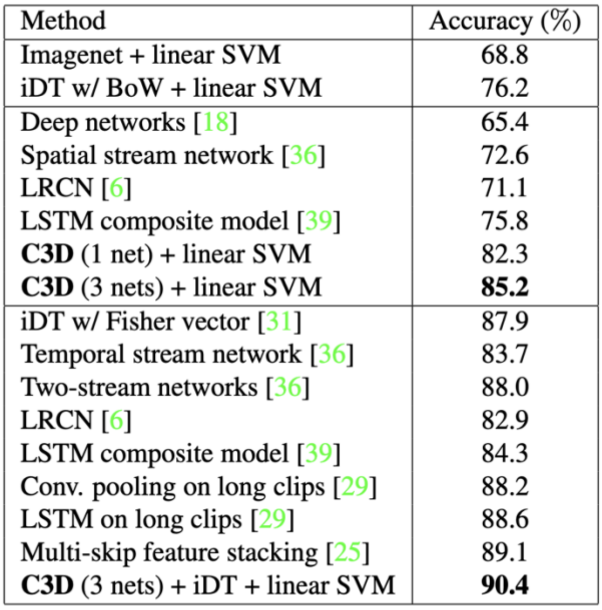
The LSTM architecture for videos was popularized in the 2014 paper *Long-term Recurrent Convolutional Networks for Visual Recognition and Description* by Donahue et. al [5]. The architecture is known as LRCN. It is a direct extension of the encoder-decoder architecture but for video representations. The strength of the LRCN network is that it can handle sequences of various lengths. It can also be adapted to other video tasks like image captioning and video description. The weakness was that the LRCN was not able to beat the state of the art at the time, however it did provide improvements over single frame architectures as noted in Table 3. Temporal modelling of spatial features is tough for a hidden recurrent layer to learn. Empirically, adding more hidden units to the RGB models did not improve past 256 hidden units. However, adding more hidden units while using Flow input yielded an accuracy boost of 1.7% from 256 units to 1024 units. This shows that the LRCN has a tough time learning optical flow or a similar representation of motion natively.



	Single Input Type		Weighted Average	
Model	RGB	Flow	1/2, 1/2	1/3, 2/3
Single frame	67.37	74.37	75.46	78.94
LRCN-fc$_6$	**68.20**	**77.28**	80.90	**82.34**

Table 3 - Activity recognition: Comparing single frame models to LRCN networks for activity recognition on the UCF101 [25] dataset, with RGB and flow inputs. [5]

3D ConvNets were established as the new state of the art in the 2015 research paper *Learning Spatiotemporal Features with 3D Convolutional Networks* by Du Tran et. al [6]. In this paper, they establish that the 3D convolution net (C3D) with a 3x3x3 kernel is the most effective in learning spatiotemporal features. Interestingly, deconvolutions reveal that the network is learning spatial appearance for the first few frames followed by salient motion in the later frames of a clip. This architecture is powerful in that many videos can be processed in real time as C3D processes at up to 313fps. The video descriptors generated by this network are also compact and discriminative as we can project the features generated by convolutions to 10 dimensions via PCA and still achieve 52.8% accuracy on the UCF101 dataset.



Method	Accuracy (%)
Imagenet + linear SVM	68.8
iDT w/ BoW + linear SVM	76.2
Deep networks [18]	65.4
Spatial stream network [36]	72.6
LRCN [6]	71.1
LSTM composite model [39]	75.8
C3D (1 net) + linear SVM	82.3
C3D (3 nets) + linear SVM	**85.2**
iDT w/ Fisher vector [31]	87.9
Temporal stream network [36]	83.7
Two-stream networks [36]	88.0
LRCN [6]	82.9
LSTM composite model [39]	84.3
Conv. pooling on long clips [29]	88.2
LSTM on long clips [29]	88.6
Multi-skip feature stacking [25]	89.1
C3D (3 nets) + iDT + linear SVM	**90.4**

Table 4 - Action recognition results on UCF101. C3D compared with baselines and state-of-the-art methods in 2015.[6]

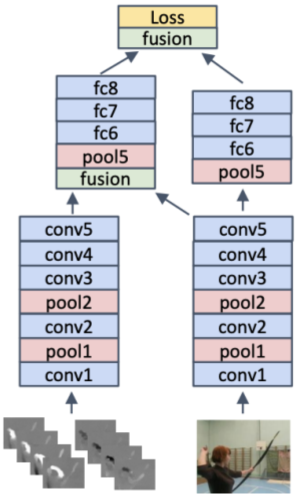
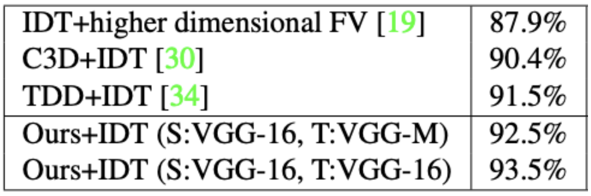


Figure 5 - Fusion architecture at two layers (after conv5 and after fc8) where both network towers are kept, one as a hybrid spatiotemporal net and one as a purely spatial network. [7]

In 2016, the focus shifted back to two stream networks. In *Convolutional Two-Stream Network Fusion for Video Action Recognition* by Zisserman et. al. [7], the authors tackled how to effectively fuse spatial and temporal data across streams and create multi-level loss that could handle long term temporal dependencies. The motivating idea here was that in order to discriminate between similar motions in different parts of the image, like brushing hair and brushing teeth, the network will need to take a combination of spatial features and motion features at a pixel location. Theoretically, methods that fuse the streams before densely connected layers could achieve this. In the proposed architecture, the authors fuse the two streams at two locations as shown in Figure 5. This network was able to better capture motion and spatial features in distinct subnetworks and beat the state of the art IDT and C3D approaches. The multi-level loss is formed by a spatiotemporal loss at the last fusion layer and a separate temporal loss that is formed from output of the temporal net. This allowed the researchers to create spatiotemporal features and model long term temporal dependencies. This method still suffers from the weaknesses of the original two stream network but performs better due to an enhanced architecture that better serves our real-world biases.



IDT+higher dimensional FV [19]	87.9%
C3D+IDT [30]	90.4%
TDD+IDT [34]	91.5%
Ours+IDT (S:VGG-16, T:VGG-M)	92.5%
Ours+IDT (S:VGG-16, T:VGG-16)	93.5%

Table 5 - Mean classification accuracy on UCF101 for approaches that use IDT features [7].

In 2017, Zhu et. al. took two stream networks a step forward by introducing a hidden stream that learns optical flow called MotionNet [8]. This end-to-end approach allowed the researchers to skip explicitly computing optical flow. This means that two streams approaches could now be real-time and errors from misprediction could also be propagated into MotionNet for more optimal optical flow features.

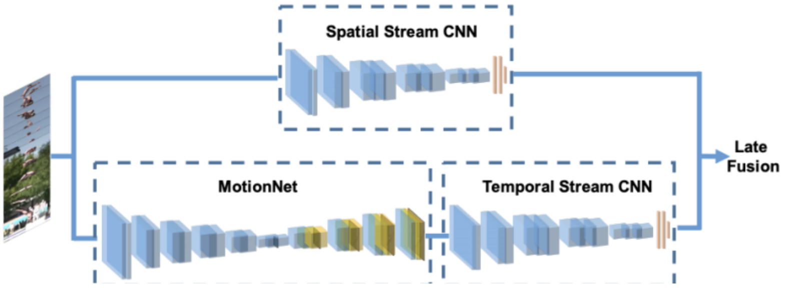
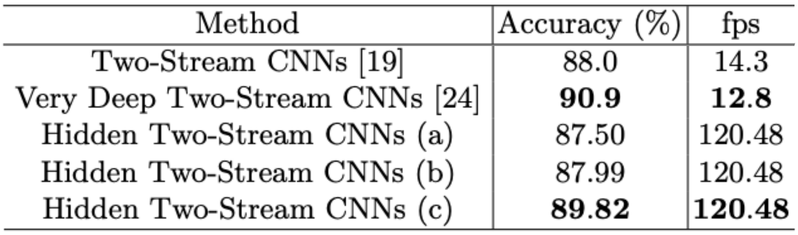


Figure 6 - MotionNet takes consecutive video frames as input and estimates motion. Then the temporal stream CNN learns to project the motion information to action labels. [8]



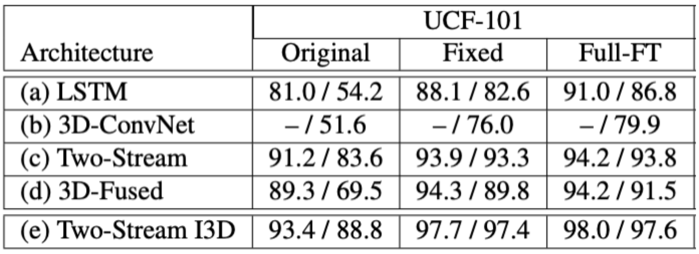
Method	Accuracy (%)	fps
Two-Stream CNNs [19]	88.0	14.3
Very Deep Two-Stream CNNs [24]	**90.9**	**12.8**
Hidden Two-Stream CNNs (a)	87.50	120.48
Hidden Two-Stream CNNs (b)	87.99	120.48
Hidden Two-Stream CNNs (c)	**89.82**	**120.48**

Table 6 - Two-stream approaches and their accuracy on UCF101. [8]

The researchers find that hidden two stream CNN's perform at a similar accuracy to non-hidden approaches but can now process up to 10x more frames per second, as seen in Table 6. This enables real-time capabilities for the two stream method.

The MotionNet subnetwork could also be extensible and applied to other deep learning methods where calculating optical flow is necessary. This is important because it allows us to make other approaches in real-time.

In 2017, *Quo Vadis, Action Recognition? A New Model and the Kinetics Dataset* by Zisserman et. al. takes C3D another step forward by merging it with learnings from two stream networks [4]. The researchers propose a novel two stream inflated 3D ConvNet (I3D). Filters and pooling kernels from 2D ConvNets are expanded into 3D, endowing them with an extra temporal dimension. This enables the researchers to take successful architectures for 2D classification and apply them to 3D. The researchers also bootstrap these 3D filters with parameters from 2D ConvNet models trained on massive image datasets like ImageNet.



Architecture	UCF-101		
	Original	Fixed	Full-FT
(a) LSTM	81.0 / 54.2	88.1 / 82.6	91.0 / 86.8
(b) 3D-ConvNet	– / 51.6	– / 76.0	– / 79.9
(c) Two-Stream	91.2 / 83.6	93.9 / 93.3	94.2 / 93.8
(d) 3D-Fused	89.3 / 69.5	94.3 / 89.8	94.2 / 91.5
(e) Two-Stream I3D	93.4 / 88.8	97.7 / 97.4	98.0 / 97.6

Table 7 - Performance on the UCF-101 and HMDB-51 test sets (split 1 of both) for architectures starting with / without ImageNet pretrained weights. Original: train on UCF-101 or HMDB-51; Fixed: features from Kinetics, with the last layer trained on UCF-101 or HMDB-51; Full-FT: Kinetics pre-training with end-to-end fine-tuning on UCF-101 or HMDB-51.[4]

Using 3D ConvNets on sequential RGB frames and sequential optical flow frames in a two stream architecture enabled the researchers to beat the state of the art on UCF101. The researchers established the clear importance of transfer learning with the use of the Kinetics dataset. Unfortunately, the model architecture they used is not end-to-end trainable and does not have real-time capabilities.

In 2017 to 2018, many advances in deep residual learning led to novel architectures like 3DResNet and pseudo-residual C3D (P3D) [9]. Unfortunately, we will not cover these papers in this literature review, but we do respectfully acknowledge their impact on the state of the art.

Most recently, in June 2019, Du Tran et. al. propose channel separated convolution networks (CSN) for the task of action recognition in *Video Classification with Channel-Separated Convolutional Networks* [10]. The researchers build on the ideas of group convolution and depth-wise convolution that received great success in Xception and MobileNet models. Fundamentally, group convolutions introduce regularization and less computations by not being fully connected. Depth-wise convolutions are the extreme case of group convolutions where the input and output channels equal the number of groups, as seen in Figure 7. Conventional convolutional networks model channel interactions and local interactions (both spatial or spatiotemporal) jointly in their 3D convolutions. The researchers propose to decompose 3x3x3 convolution kernels into two distinct layers, where the first layer is a

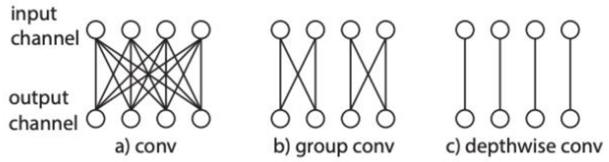


Figure 7 - (a) A conventional convolution, which has only one group. (b) A group convolution with 2 groups. (c) A depthwise convolution where the number of groups matches the number of input/output filters.

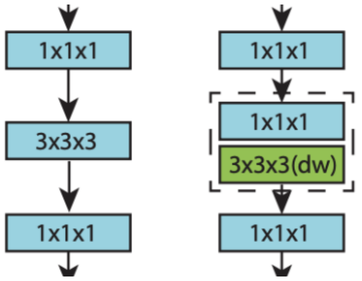
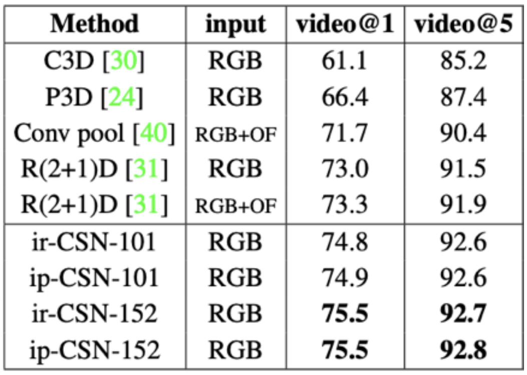


Figure 8 - (a) A standard ResNet bottleneck block. (b) An interaction preserved bottleneck block

1x1x1 convolution for local channel interaction and the second layer is a 3x3x3 depth-wise convolution for local spatiotemporal interactions. By using these blocks, the researchers significantly decrease the number of parameters in the network and introduce a strong form of regularization. The channel separated blocks allow for the network to locally learn spatial and spatiotemporal features in distinct layers. As shown in Table 8, The CSN improves on state of the art RGB methods like R(2+1)D, C3D, and P3D on the Sports-1M dataset. The network is also 2-4x faster during inference. The model is also trained from scratch, where the rest of the models in the table are pretrained on ImageNet or Kinetics dataset. This novel architecture improves on previous factorized networks while reducing overfitting, being exceptionally fast, and producing state of the art accuracy on benchmark datasets.

Analysis

The current state of the art for action recognition is the channel separated network. This network effectively captures spatial and spatiotemporal features in their own distinct layers. The channel separated convolution blocks learns these features distinctly but combines them locally at all stages of convolution. This alleviates the need to perform slow fusion of temporal and spatial two stream networks. The network also does not need to decide between learning spatial or temporal features as in C3D where the network can decide to learn features that are mixed between the two dimensions. This network effectively captures the bias that 2D spatial slices should form a proper image, whereas a 2D slice in the temporal direction should make very little sense. In this way, the researchers enforce this bias by creating two separate distinct layers to process each direction. Channel separation is an important step forward in action recognition and has beat state of the art results even when trained from scratch. It is also capable of real time inference. For these reasons, we believe CSN's are the current state of the art.



Method	input	video@1	video@5
C3D [30]	RGB	61.1	85.2
P3D [24]	RGB	66.4	87.4
Conv pool [40]	RGB+OF	71.7	90.4
R(2+1)D [31]	RGB	73.0	91.5
R(2+1)D [31]	RGB+OF	73.3	91.9
ir-CSN-101	RGB	74.8	92.6
ip-CSN-101	RGB	74.9	92.6
ir-CSN-152	RGB	**75.5**	**92.7**
ip-CSN-152	RGB	**75.5**	**92.8**

Table 8 - Comparisons with state-of-the-art architectures on Sports-1M

Conclusion

We have learned that deep learning has revolutionized the way we process videos for action recognition. Deep learning literature has come a long way from using improved Dense Trajectories. Many learnings from the sister problem of image classification has been used in advancing deep networks for action recognition. Specifically, the usage of convolution layers, pooling layers, batch normalization, and residual connections have been borrowed from the 2D space and applied in 3D with substantial success. Many models that use a spatial stream are pretrained on extensive image datasets. Optical flow has also had an important role in representing temporal features in early deep video architectures like the two stream networks and fusion networks. Optical flow is our mathematical definition of how we believe movement in subsequent frames can be described as densely calculated flow vectors for all pixels. Originally, networks bolstered performance by using optical flow. However, this made networks unable to be end-to-end trained and limited real-time capabilities. In modern deep learning, we have moved past optical flow and we instead architect networks that are able to natively learn temporal embeddings and are end-to-end trainable.

We have also learned that action recognition is a truly unique problem with its own set of complications. The first source of friction is the high computation and memory cost associated with 3D convolutions. Some models take over 2 months to train on Sports-1M on modern

GPU’s. The second source of friction is that there is no standard benchmark for video architecture search [11]. Sports-1M and UCF101 are highly correlated and false-label assignment is common when a portion of a video is selected to be trained on but actually may not contain the actual action as it may be in another part of the video. The last source of friction is that designing a video deep neural network is nontrivial. The choice of layers, how to preprocess the input, and how to model the temporal dimension is an open problem. The authors of the papers above attempt to tackle these issues in an empirical fashion and propose novel architectures that resolve temporal modelling in videos.

For future research, we recommend looking into how to include more biases we have of the real world in deep video network architecture. An interesting vertical to study is how depth modelling can relate to better video classifications. Current approaches to video classification have to learn that the videos are taken in a 3D environment. Depth forms an important part of our spatial perception. It could be that current approaches have to learn how to express depth in their spatiotemporal modelling of 2D features. Perhaps using monocular depth estimation networks can aid the current video networks in creating a better understanding of the environment itself. An important observation is that any spatial changes in a video come from two sources: a transformation of an external object we are observing, or the observer itself changing viewing angle or position. Both these sources of movement have to be learned by the current networks. It would be interesting to investigate how depth fields could be used to model either sources of change.

Citations

[1] Heng Wang, Alexander Kläser, Cordelia Schmid, Liu Cheng-Lin. Action Recognition by Dense Trajectories. CVPR 2011 - IEEE Conference on Computer Vision

[2] Karpathy, Andrej, et al. "Large-scale video classification with convolutional neural networks." Proceedings of the IEEE conference on Computer Vision and Pattern Recognition. 2014.

[3] Simonyan, Karen, and Andrew Zisserman. "Two-stream convolutional networks for action recognition in videos." Advances in neural information processing systems. 2014.

[4] Carreira, Joao, and Andrew Zisserman. "Quo vadis, action recognition? a new model and the kinetics dataset." proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2017.

[5] Donahue, Jeffrey, et al. "Long-term recurrent convolutional networks for visual recognition and description." Proceedings of the IEEE conference on computer vision and pattern recognition. 2015.

[6] Tran, Du, et al. "Learning spatiotemporal features with 3d convolutional networks." Proceedings of the IEEE international conference on computer vision. 2015.

[7] Feichtenhofer, Christoph, Axel Pinz, and Andrew Zisserman. "Convolutional two-stream network fusion for video action recognition." Proceedings of the IEEE conference on computer vision and pattern recognition. 2016.

[8] Zhu, Yi et al. “Hidden Two-Stream Convolutional Networks for Action Recognition.” Lecture Notes in Computer Science (2019): 363–378. Crossref. Web.

[9] Qiu, Zhaofan, Ting Yao, and Tao Mei. "Learning spatio-temporal representation with pseudo-3d residual networks." proceedings of the IEEE International Conference on Computer Vision. 2017.

[10] Tran, Du, et al. "Video Classification with Channel-Separated Convolutional Networks." arXiv preprint arXiv:1904.02811 (2019).

[11] Tran, Du, et al. "Convnet architecture search for spatiotemporal feature learning." arXiv preprint arXiv:1708.05038 (2017).