Dual Stream Spatio-Temporal Motion Fusion With Self-Attention For Action Recognition

Md Asif Jalal[a], Waqas Aftab[b], Roger K Moore[a], Lyudmila Mihaylova[b]

[a]Department of Computer Science, University of Sheffield, S1 4DP, UK
[b]Department of Automatic Control and Systems Engineering, University of Sheffield, S1 3JD, UK
{majalal1,waftab1,r.k.moore,l.s.mihaylova}@sheffield.ac.uk

Abstract—**Human action recognition in diverse and realistic environments is a challenging task. Automatic classification of action and gestures has a significant impact on human-robot interaction and human-machine interaction technologies. Due to the prevalence of complex real-world problems, it is non-trivial to produce a rich representation of actions and to produce an effective categorical distribution of large action classes. Deep convolutional neural networks have obtained great success in this area. Many researchers have proposed deep neural architectures for action recognition while considering the spatial and temporal aspects of the action. This research proposes a dual stream spatio-temporal fusion architecture for human action classification. The spatial and temporal data is fused using an attention mechanism. We investigate two fusion techniques and show that the proposed architecture achieves accurate results with much fewer parameters as compared to the traditional deep neural networks. We achieved** 99.1 **% absolute accuracy on the UCF-101 test set.**

Index Terms—**Action recognition, attention networks, fusion, deep neural networks**

I. Introduction

Human activity recognition in a real-world environment is gaining popularity for its various applications in day to day life. It aims to classify human actions by a series of observations of human actions at a given period. There are numerous applications of action recognition, such as human-robot interaction, wearable technologies, surveillance, multimedia content annotation and measuring similarity. The temporal, motion and contextual aspect of a video makes it different from standard image classification. The spatio-temporal feature representation and generalisation are non-trivial due to the real-world obstacles, such as jitter, lighting conditions, camera viewpoint changes and camera motion.

More researches have been published on this problem so far. Gaussian mixture models, SVM models and probabilistic models were proposed using hand-crafted features [1–5]. However, deep neural models become very popular because they can generate high-level features from low level features [6–8]. Initially, the deep neural architectures do not perform exceedingly compare to the traditional hand-crafted feature based methods [9]. Deep convolutional neural architectures are also introduced for vision-based action recognition problems [10–12]. Since then different variants of convolutional neural networks (CNN) are introduced exploring spatial and temporal modelling [13, 14].

This huge revolution in action recognition research also evolved the experimental data. From stationary camera and controlled environment oriented [15] action database, the research community, towards more in the wild and real-world oriented database [16–18].

In this paper, we investigate the spatiotemporal relationship between the sequences in a video for action recognition. We adopt an attention mechanism [19] in our framework. Our contribution is two-fold

i. We propose a deep neural net framework that performs with high accuracy (state-of-the-art with UCF-101) with relatively fewer parameters compare to the current state-of-the-art architectures.
ii. We investigate fusion between the spatial and temporal channels.

The paper is organised as follows. Section 2 discusses the previous work related to this research, section 3 explains our approach and the basic building blocks of the proposed frameworks, section 4 describes the proposed frameworks, section 5 describes the experimental scenarios and interprets the results. Finally, section 5 concludes the paper and proposes future research path.

II. Related Works

The CNN based methods, are applied for image and video processing, require minimum pre-processing. Karpathy et al. proposed CNN models for video classification with large databases [11]. Moreover, the feature extraction and classification tasks can be solved simultaneously by the network. These methods have provided promising results in the field of computer vision [20], machine learning and pattern recognition. Various implementations of CNN networks have been proposed for action recognition [21, 22]. The 3D CNN features based action recognition was proposed in [12, 13].A two stream, a spatial and a temporal stream based approach have been proposed for action recognition in [10]. An Attention-based Temporal Weighted CNN (ATW) combines a visual attention model with a temporal weighted multi-stream CNN [23].

Recurrent neural networks (RNN) have been achieved good results in temporal modelling of sequential data [24]. A visual action is a sequence of consecutive events happens in a period of time. Modelling temporal context and modelling the relation between those sequences can give a rich representation. Visual sequence modelling has been carried out by several in the literature [25, 26]. Veerial et al. [27] show that the salient motion feature between the consecutive frames (derivative of states between frames) can be used successfully with long-short-term-memory networks (LSTMs) [24] to model time-series action sequence modelling.

Vaswani et al. [28] propose an attention mechanism based architecture with feed-forward neural networks to show that dependencies in between the elements in a sequence can be learned by attention mechanism. Attention networks have become vastly popular for modelling long term dependencies [28–32]. Wang et. al [19] propose non-local networks to measure positional dependency within the same sequence.

III. Approach

In this paper, our motivation is to investigate the dependency between spatial and temporal data for action recognition. The dual stream architecture [10] has been adopted for the base of our framework. This architecture is a computational model of the two-stream hypothesis [33, 34] which states that human visual cortex system consists of dual channels (dorsel and ventral) to process spatial and temporal information. We will use RGB frames for the spatial modelling and dense optical flow for temporal modelling.

A. Optical Flow

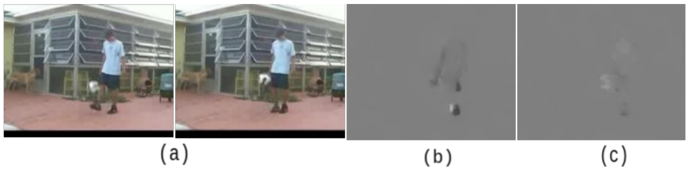


Fig. 1: (a) consecutive pair of images (b) and (c) horizontal and vertical component of optical flow

Optical Flow (OF) is a visual object tracking method that approximates the relative motion of an object and the observer (sensor). The OF algorithm assumes constant pixel intensity across consecutive frames and relatively small object motion (displacement). Based on these assumptions, the OF algorithm calculates a vector displacement field around each pixel for 2D tracking and each voxel for 3D tracking. Various techniques have been proposed to determine the OF such as horn-schunck [35], lucas-kanade [36] and brox [37]. In this paper, we have used the brox [37] method to extract OF. The OF is a displacement vector d_t between consecutive frames t and $t+1$. The vector $d_t(u,v)$ is the displacement of point (u,v) from frame t to $t+1$. The horizontal u_t^x, u_{t+1}^x and vertical v_t^y, v_{t+1}^y are used as input channels in the CNN for temporal modelling. A sample OF frame is shown in figure 1.

B. Convolutional Neural Network

Convolutional Neural Networks (CNN), like other neural networks, are multi-layer neural networks. A CNN consists of the convolution and other layers (such as sub-sampling, pooling, ReLU, fully connected, loss) working in a deep learning framework. The initial layers detect low-level features, and the last layers work on the high-level feature space. The characteristics of these networks are that each feature of the layer is connected to a local area, also called *local receptive field*, of the previous layer. These areas are overlapping and, when combined, provide an overall result for a given task. The feature maps are convoluted with kernels $f[x_b, y_b]$. The kernels are the local receptive field $f[x_a, y_a]$. While the kernels slide through the image, they extract visual features (edges, corners or more abstract features) and combine the set of outputs to form feature maps. If the kernels of size $[h \times w \times N]$ ([height $\times$ width $\times$ depth], and $n = 1, 2, \cdots, N$) are used, the n^{th} convolutional feature map would be:

$$y_n = f\left(\sum_j g_n * x_j\right), \tag{1}$$

where g_n is the n^{th} kernel and x_j $(j = 1, 2, \cdots, J)$ is the j^{th} input feature map of size $[A \times B]$ and $f(\cdot)$ is a nonlinear activation function. The kernel size is significant for preserving locality in the whole network as well as controlling representations [38].

In this research, we use a smaller kernel size (2-5) to preserve locality by considering a small neighbourhood at a time. Smaller kernel sizes can increase non linearity in the network and enable feature fusion [39, 40]. According to some research, stacking more than one CNN layer results in increased non-linearity and richer representations [41]. Generally, the CNN layer is followed by a pooling layer. The pooling layer improves the discriminability power of the network and robustness to shift and distortions [42]. This means, pooling brings invariance to the network. However, it is crucial to control the kernel size in pooling to keep it from losing information. The network learns faster with decorrelated network parameters have zero means and unit variances [42]. Loffee et al. [43] proposed batch normalisation for approaching this issue with reducing internal covariance shift in a batch. We adapted these methods in our proposed framework.

C. Self Attention Networks

Self-attention networks have the flexibility of modelling long-term inter-sequence dependencies. In this research, we use self-attention as non-local networks [19, 44] to model the relationships between the regions in the feature maps from previous layers. The mechanism is shown in figure 2. The features from previous CNN layers $\mathbf{y}$ are transformed into three feature spaces $\mathbf{j}$, $\mathbf{k}$ and $\boldsymbol{l}$, where

$$j\left(y\right) = W_j y \qquad k\left(y\right) = W_k y \qquad l\left(y\right) = W_l y \tag{2}$$

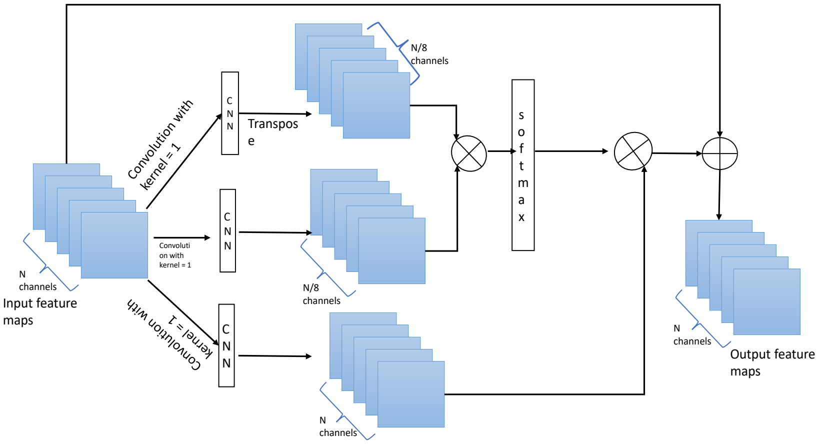


Fig. 2: Convolutional self-attention mechanism.

Here W_j, W_k and W_l are network weights learned through back-propagation. The number of channels in W_j, W_k is less than the number of channels in the features. However, W_k has the same number of channels as input feature y. Dot product is used to calculate the relationship between j and k. Then we normalize it using the $softmax$ function.

$$e_{ij} = softmax\left(j\left(y_i\right)^T k\left(y_j\right)\right) \tag{3}$$

The attention map is calculated by doing matrix multiplication between e and $l(y)$. A scaling factor is multiplied with the attention map and the result is added with the input feature map.

$$attention\_output = \gamma(el(y)) + y \tag{4}$$

In this work, γ is randomly initialized. This layer learns the non-local dependencies as well as the local neighbourhood.

IV. The Proposed Framework

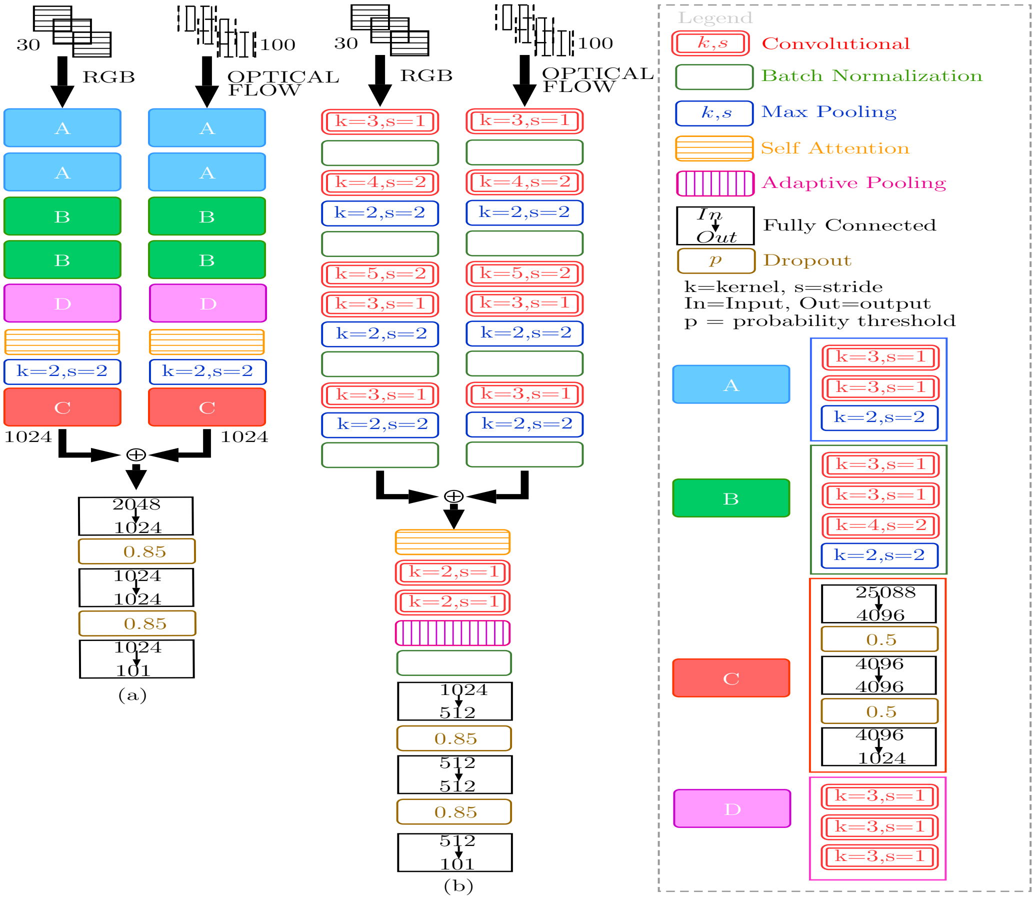
Our contribution is two-fold for the proposed architecture. We propose two models based on two different fusion techniques. The first model shows late fusion and the second model shows early fusion. The frameworks are shown in figure 3. Each video V is divided into N frames $\{f_1, f_2, ..., f_N\}$. Consecutive frames are highly redundant, we choose frames sequentially but having small time distance with each other.

1) Late Fusion: We take inspiration from VGGNET [45] for the late fusion model. The spatial stream operates on a sequence of RGB video frames. The frames are stacked and fused by interpreting each of the frames as an individual channel. The model consists of 5 different CNN layer blocks. Block A has two convolution layers with kernel size 3, followed by a maxpooling layer with kernel size 2. Block B has three convolution layers with kernel size $3, 3, 4$ respectively, followed by a maxpooling layer with kernel size 2. Block D has three convolution layers with kernel size 3. This followed by a convolutional attention layer. The attention layer (III-C) fuses the spatial feature maps from block D of channels sized 128.The convolution layers in the attention layer have a kernel size of 1. Block C has three fully connected layers. We have used dropout for regularisation. The temporal stream has a similar architecture as shown in figure 3. The output o_r of spatial o_s and temporal streams o_t are fused using concatenation

$$o_r = o_s \oplus o_t, \tag{5}$$

(where $\oplus$ denotes concatenation). Then o_r is fed to fully connected layers and *dropout* is used for regularization.

2) Early Fusion: The proposed early fusion model has a simplistic approach. Both the spatial and temporal stream have a smaller feature extraction layer compared to the previous model. The input channels are fed to a convolution layer with kernel size 3 followed by a batch normalisation layer. The number of output channels is 128. The feature maps are slowly down-sampled and then up-sampled in the following layers. Maxpooling has been used to introduce sparsity in the network parameters. After each series of convolution process, we use batch normalisation extensively to reduce correlation among the parameters at the same layer within the network. Both spatial and temporal stream produce 128 channels of feature maps. These feature maps are fused using a self-attention layer. The network weights W_j, W_k, W_l, mentioned in III-C, are three convolutional layers with kernel size 1. The number of output channels for W_j and W_k is one-eighth of the input channels (256) in the self-attention layer. The output channels are decreased to reduce the computation time. The number of output channels for W_l is the same as the input channels in the self-attention layer. The scaling factor γ is randomly initialised. The kernel size in the attention layer is set to 1 to perform feature level fusion. The output feature maps from the attention layer are fed in an adaptive pooling layer to produce fixed sized output feature maps. These feature maps are given



(a)

(b)

Fig. 3: Early and late fusion with attention network

as input to three fully connected layers. Similar to the previous model, *dropout* has been used as a regularizer.

V. Performance Evaluation

A. Dataset

In this paper, the UCF-101 dataset [46] is used for evaluating the performance of the proposed method. This dataset consists of 101 actions annotated for 13320 Youtube video clips. This dataset is among the biggest (in terms of the number of classes and videos) publicly available and annotated datasets to date. The videos have been uploaded by non-professionals which include additional challenges such as shaking cameras, inconsistent viewpoints and changing resolutions. Moreover, there are groups of classes which are quite similar to each other such as violin and cello playing.

B. Experiment

The experiments have been conducted using the PyTorch [47] deep learning framework. We used 10 motion frames with interval for the RGB stream training. In the training phase, the frame sequences are change in every 100th epoch. For example, if we use $f_5, f_{15}, f_{25}, f_{35}, ..., f_{95}$ frames in the the first 100 epochs, we change the sequence to $f_8, f_{18}, f_{28}, ..., f_{98}$ frames for 100 to 200th epoch. One Nvidia GTX 1080ti GPU has been used for executing the experiments.

1) Learning: The adam optimiser [48] was applied to mini-batch of 25 videos with categorical cross-entropy loss. The momentum and weight decay are set to 0.9 and 0 respectively. Throughout the network, the learning rate was set to 0.0001. To prevent the network from over-fitting dropout layers were used with the fully connected layers. In the late fusion model, figure 3, the dropout rate in block C was set to 0.5 but after the fusion, the dropout was set to 0.85 in the classifier section.

2) Data Augmentation: Random horizontal flipping and random cropping were applied to the frames for data augmentation to increase the diversity of the training samples. The frames are normalised and re-sized to [224 × 224] images.

TABLE I: Comparison of the proposed methods with other state-of-the-art methods in terms of accuracy (%)

Method	**Accuracy (%)**
Two Stream [10]	88.0
C3D (3 nets) [12]	85.2
Two stream + LSTM [49]	88.6
Two stream VGGNet-16	90.9
Long Term Temporal Convolution [50]	91.7
KVMF [51]	93.1
TSN 3 Modaliites [14]	94.2
Proposed Network I(late fusion)	**98.8**
Proposed Network II(early fusion)	**99.1**

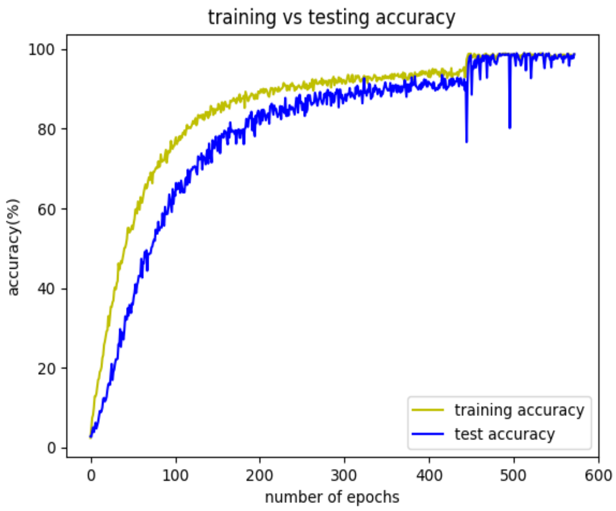


Fig. 4: Training vs Testing accuracy (%) during early fusion model training

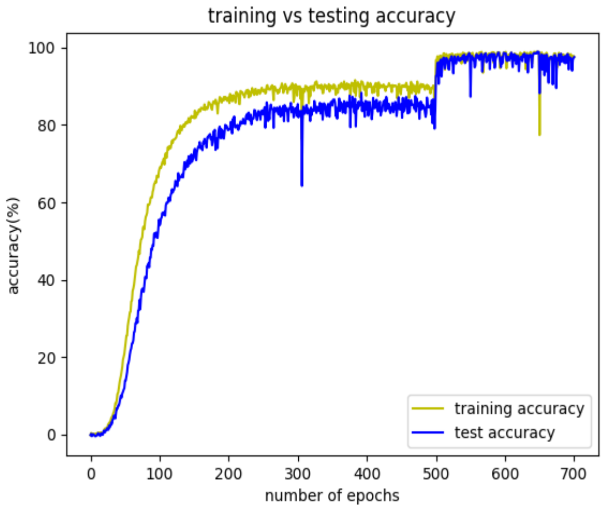


Fig. 5: Training vs Testing accuracy (%) during late fusion model training

C. Results

This section presents the results and their comparison with the state-of-the-art. The proposed frameworks have been evaluated on UCF-101 dataset with the split I. The accuracy comparisons with the state-of-the-art systems are shown in table I. It can be clearly seen that both the proposed frameworks achieved state-of-the art results in UCF-101 test dataset. We demonstrated two types of model late fusion attention model and early fusion attention model. With the late fusion attention model, we explored very deep neural architecture, and it has 280314599 parameters. The early fusion model has 4956442 parameters, which is around 98% smaller than the late fusion network. The training vs testing accuracy is shown for both of the models in figure 4 and figure 5.

D. Discussion

Two points can be clearly drawn from figure 4 and figure 5. Firstly, the representation learning on training data is truly generalized for the test data. Secondly, from the initial training stage, both the networks do not over-fit. Furthermore, in this research, we used the training videos in such a way that most of the frames are utilised. As mentioned in section IV, every 100 th epoch we changed the frame sequences but we kept the frame order intact. This allows us to use the maximum training data as well as provide good data augmentation.

We can clearly say that the early fusion attention model converges faster than the late fusion attention model. Despite having 98% smaller size than the late fusion attention model, the early fusion attention model performs exceptionally well. Also, the early-fusion-attention model has fewer parameters than the popular deep neural networks, such as VGGNet, AlexNet, ResNet.

VI. Conclusion

We presented two dual-stream attention fusion frameworks. The dual-stream learning is to model the dorsal and ventral stream learning hypotheses for human cognition. We investigated two stages of fusion in between those streams. Also, a reduction of the computational cost with the early fusion attention model is seen which has a smaller network size without compromising the performance. This research brought the state-of-the-art to a different level as explained in section V-D. Finally, this work has tried to bring together good practices for designing CNN and deep networks. The future research path will be examining the proposed models with bigger databases with more diverse and complex categories.

References

[1] I. Laptev and T. Lindeberg, “Space-time interest points,” in *Proceedings of the Ninth IEEE International Conference on Computer Vision - Volume 2*, ser. ICCV ’03. Washington, DC, USA: IEEE Computer Society, 2003, pp. 432–. [Online]. Available: http://dl.acm.org/citation.cfm?id=946247.946605

[2] I. Laptev, M. Marszalek, C. Schmid, and B. Rozenfeld, “Learning realistic human actions from movies,” *IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1–8, 2008.

[3] M. Vrigkas, V. Karavasilis, C. Nikou, and I. A. Kakadiaris, “Matching mixtures of curves for human action recognition,” *Comput. Vis. Image Underst.*, vol. 119, pp. 27–40, Feb. 2014. [Online]. Available: http://dx.doi.org/10.1016/j.cviu.2013.11.007

[4] Y. Zhu, W. Chen, and G. Guo, “Evaluating spatiotemporal interest point features for depth-based action recognition,” *Image Vision Comput.*, vol. 32, no. 8, pp. 453–464, Aug. 2014. [Online]. Available: http://dx.doi.org/10.1016/j.imavis.2014.04.005

[5] A. Fathi and G. Mori, “Action recognition by learning mid-level motion features,” in *2008 IEEE Conference on Computer Vision and Pattern Recognition*, June 2008, pp. 1–8.

[6] Y. Bengio and Y. Lecun, *Scaling learning algorithms towards AI*. MIT Press, 2007.

[7] Y. Bengio, “Learning deep architectures for ai,” *Found. Trends Mach. Learn.*, vol. 2, no. 1, pp. 1–127, Jan. 2009. [Online]. Available: http://dx.doi.org/10.1561/2200000006

[8] G. E. Hinton and R. R. Salakhutdinov, “Reducing the dimensionality of data with neural networks,” *Science*, vol.

313, no. 5786, pp. 504–507, 2006. [Online]. Available: http://science.sciencemag.org/content/313/5786/504

[9] X. Peng, L. Wang, X. Wang, and Y. Qiao, "Bag of visual words and fusion methods for action recognition: Comprehensive study and good practice," *CoRR*, vol. abs/1405.4506, 2014. [Online]. Available: http://arxiv.org/abs/1405.4506

[10] K. Simonyan and A. Zisserman, "Two-stream convolutional networks for action recognition in videos," in *Advances in neural information processing systems*, 2014, pp. 568–576.

[11] A. Karpathy, G. Toderici, S. Shetty, T. Leung, R. Sukthankar, and L. Fei-Fei, "Large-scale video classification with convolutional neural networks," in *Proceedings of the 2014 IEEE Conference on Computer Vision and Pattern Recognition*, ser. CVPR '14. Washington, DC, USA: IEEE Computer Society, 2014, pp. 1725–1732. [Online]. Available: https://doi.org/10.1109/CVPR.2014.223

[12] D. Tran, L. D. Bourdev, R. Fergus, L. Torresani, and M. Paluri, "C3D: generic features for video analysis," *CoRR*, vol. abs/1412.0767, 2014. [Online]. Available: http://arxiv.org/abs/1412.0767

[13] S. Ji, W. Xu, M. Yang, and K. Yu, "3d convolutional neural networks for human action recognition," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 35, no. 1, pp. 221–231, Jan 2013.

[14] L. Wang, Y. Xiong, Z. Wang, Y. Qiao, D. Lin, X. Tang, and L. Van Gool, "Temporal segment networks: Towards good practices for deep action recognition," in *Computer Vision – ECCV 2016*, B. Leibe, J. Matas, N. Sebe, and M. Welling, Eds. Cham: Springer International Publishing, 2016, pp. 20–36.

[15] C. Schuldt, I. Laptev, and B. Caputo, "Recognizing human actions: A local svm approach," in *Proceedings of the Pattern Recognition, 17th International Conference on (ICPR'04) Volume 3 - Volume 03*, ser. ICPR '04. Washington, DC, USA: IEEE Computer Society, 2004, pp. 32–36. [Online]. Available: http://dx.doi.org/10.1109/ICPR.2004.747

[16] K. Soomro, A. R. Zamir, and M. Shah, "UCF101: A dataset of 101 human actions classes from videos in the wild," *CoRR*, vol. abs/1212.0402, 2012. [Online]. Available: http://arxiv.org/abs/1212.0402

[17] A. Karpathy, G. Toderici, S. Shetty, T. Leung, R. Sukthankar, and L. Fei-Fei, "Large-scale video classification with convolutional neural networks," in *CVPR*, 2014.

[18] S. Abu-El-Haija, N. Kothari, J. Lee, A. P. Natsev, G. Toderici, B. Varadarajan, and S. Vijayanarasimhan, "Youtube-8m: A large-scale video classification benchmark," in *arXiv:1609.08675*, 2016. [Online]. Available: https://arxiv.org/pdf/1609.08675v1.pdf

[19] X. Wang, R. B. Girshick, A. Gupta, and K. He, "Non-local neural networks," *CoRR*, vol. abs/1711.07971, 2017. [Online]. Available: http://arxiv.org/abs/1711.07971

[20] A. Sharif Razavian, H. Azizpour, J. Sullivan, and S. Carlsson, "Cnn features off-the-shelf: an astounding baseline for recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition workshops*, 2014, pp. 806–813.

[21] G. W. Taylor, R. Fergus, Y. LeCun, and C. Bregler, "Convolutional learning of spatio-temporal features," in *European conference on computer vision*. Springer, 2010, pp. 140–153.

[22] Y. Tas and P. Koniusz, "Cnn-based action recognition and supervised domain adaptation on 3D body skeletons via kernel feature maps," *arXiv preprint arXiv:1806.09078*, 2018.

[23] J. Zang, L. Wang, Z. Liu, Q. Zhang, G. Hua, and N. Zheng, "Attention-based temporal weighted convolutional neural network for action recognition," in *Proceedings of the IFIP International Conference on Artificial Intelligence Applications and Innovations*. Springer, 2018, pp. 97–108.

[24] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Comput.*, vol. 9, no. 8, pp. 1735–1780, Nov. 1997. [Online]. Available: http://dx.doi.org/10.1162/neco.1997.9.8.1735

[25] A. Karpathy and F. Li, "Deep visual-semantic alignments for generating image descriptions," *CoRR*, vol. abs/1412.2306, 2014. [Online]. Available: http://arxiv.org/abs/1412.2306

[26] J. Donahue, L. A. Hendricks, S. Guadarrama, M. Rohrbach, S. Venugopalan, K. Saenko, and T. Darrell, "Long-term recurrent convolutional networks for visual recognition and description," *CoRR*, vol. abs/1411.4389, 2014. [Online]. Available: http://arxiv.org/abs/1411.4389

[27] V. Veeriah, N. Zhuang, and G.-J. Qi, "Differential recurrent neural networks for action recognition," in *The IEEE International Conference on Computer Vision (ICCV)*, December 2015.

[28] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin, "Attention is all you need," *CoRR*, vol. abs/1706.03762, 2017. [Online]. Available: http://arxiv.org/abs/1706.03762

[29] K. Xu, J. Ba, R. Kiros, K. Cho, A. Courville, R. Salakhudinov, R. Zemel, and Y. Bengio, "Show, attend and tell: Neural image caption generation with visual attention," in *Proceedings of the 32nd International Conference on Machine Learning*, ser. Proceedings of Machine Learning Research, F. Bach and D. Blei, Eds., vol. 37. Lille, France: PMLR, 07–09 Jul 2015, pp. 2048–2057. [Online]. Available: http://proceedings.mlr.press/v37/xuc15.html

[30] M. Jaderberg, K. Simonyan, A. Zisserman, and k. kavukcuoglu, "Spatial transformer networks," in *Advances in Neural Information Processing Systems 28*, C. Cortes, N. D. Lawrence, D. D. Lee, M. Sugiyama, and R. Garnett, Eds. Curran Associates, Inc., 2015, pp. 2017–2025. [Online]. Available: http://papers.nips.cc/paper/5854-spatial-transformer-networks.pdf

[31] J. Ba, V. Mnih, and K. Kavukcuoglu, "Multiple object recognition with visual attention," *CoRR*, vol. abs/1412.7755, 2015.

[32] S. Woo, J. Park, J. Lee, and I. S. Kweon, "CBAM: convolutional block attention module," *CoRR*, vol. abs/1807.06521, 2018. [Online]. Available: http://arxiv.org/abs/1807.06521

[33] M. A. Goodale and A. D. Milner, "Separate visual pathways for perception and action," 1992.

[34] A. Sedda and F. Scarpina, "Dorsal and ventral streams across sensory modalities," *Neuroscience Bulletin*, vol. 28, no. 3, pp. 291–300, Jun 2012. [Online]. Available: https://doi.org/10.1007/s12264-012-1223-9

[35] B. K. Horn and B. G. Schunck, "Determining optical flow," *Artificial intelligence*, vol. 17, no. 1-3, pp. 185–203, 1981.

[36] B. D. Lucas, T. Kanade *et al.*, "An iterative image registration technique with an application to stereo vision," 1981.

[37] T. Brox, A. Bruhn, N. Papenberg, and J. Weickert, "High accuracy optical flow estimation based on a theory for warping," in *ECCV*, 2004.

[38] Y. LeCun, Y. Bengio, and G. E. Hinton, "Deep learning," *Nature*, vol. 521, no. 7553, pp. 436–444, 2015. [Online]. Available: https://doi.org/10.1038/nature14539

[39] M. Lin, Q. Chen, and S. Yan, "Network in network," *CoRR*, vol. abs/1312.4400, 2013. [Online]. Available: http://arxiv.org/abs/1312.4400

[40] S. H. HasanPour, M. Rouhani, M. Fayyaz, M. Sabokrou, and E. Adeli, "Towards principled design of deep convolutional networks: Introducing simpnet," *CoRR*, vol. abs/1802.06205, 2018. [Online]. Available: http://arxiv.org/abs/1802.06205

[41] D. H. Wolpert, "Stacked generalization," *Neural Networks*, vol. 5, pp. 241–259, 1992.

[42] Y. LeCun, P. Haffner, L. Bottou, and Y. Bengio, "Object recognition with gradient-based learning," in *Shape, Contour and Grouping in Computer Vision*. London, UK, UK: Springer-Verlag, 1999, pp. 319–. [Online]. Available: http://dl.acm.org/citation.cfm?id=646469.691875

[43] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating

deep network training by reducing internal covariate shift," *CoRR*, vol. abs/1502.03167, 2015. [Online]. Available: http://arxiv.org/abs/1502.03167

[44] H. Zhang, I. Goodfellow, D. Metaxas, and A. Odena, "Self-Attention Generative Adversarial Networks," *arXiv e-prints*, p. arXiv:1805.08318, May 2018.

[45] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *CoRR*, vol. abs/1409.1556, 2014. [Online]. Available: http://arxiv.org/abs/1409.1556

[46] K. Soomro, A. R. Zamir, and M. Shah, "Ucf101: A dataset of 101 human actions classes from videos in the wild," *arXiv preprint arXiv:1212.0402*, 2012.

[47] A. Paszke, S. Gross, S. Chintala, G. Chanan, E. Yang, Z. DeVito, Z. Lin, A. Desmaison, L. Antiga, and A. Lerer, "Automatic differentiation in pytorch," in *NIPS-W*, 2017.

[48] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *CoRR*, vol. abs/1412.6980, 2014. [Online]. Available: http://arxiv.org/abs/1412.6980

[49] J. Y. Ng, M. J. Hausknecht, S. Vijayanarasimhan, O. Vinyals, R. Monga, and G. Toderici, "Beyond short snippets: Deep networks for video classification," *CoRR*, vol. abs/1503.08909, 2015. [Online]. Available: http://arxiv.org/abs/1503.08909

[50] G. Varol, I. Laptev, and C. Schmid, "Long-term temporal convolutions for action recognition," *CoRR*, vol. abs/1604.04494, 2016. [Online]. Available: http://arxiv.org/abs/1604.04494

[51] W. Zhu, J. Hu, G. Sun, X. Cao, and Y. Qiao, "A key volume mining deep framework for action recognition," *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1991–1999, 2016.