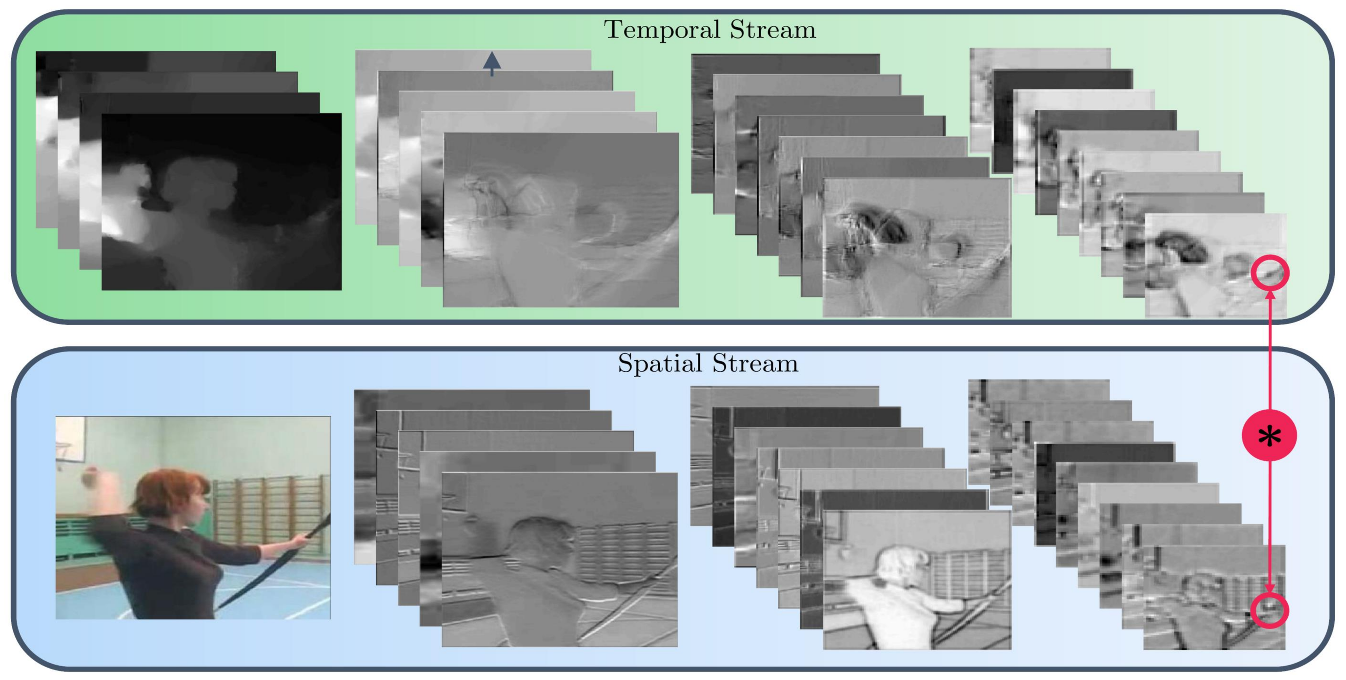
Convolutional Two-Stream Network Fusion for Video Action Recognition

Christoph Feichtenhofer — feichtenhofer@tugraz.at
Axel Pinz — axel.pinz@tugraz.at
Andrew Zisserman — az@robots.ox.ac.uk

Graz University of Technology
University of Oxford

Code & models available at **github.com/feichtenhofer/twostreamfusion**

1. Overview



- We study a number of ways of fusing ConvNet towers both spatially and temporally

We make the following findings:

1. A spatial and temporal network can be fused at a convolution layer with a substantial saving in parameters
2. It is best to fuse such networks at the last convolutional layer
3. Pooling of abstract conv features over spatiotemporal neighbourhoods further boosts performance

- Based on our studies we propose a new ConvNet architecture

2. How to fuse the two streams spatially?

- Fuse the two networks such that channel responses at the same pixel position are put in correspondence

Input features $\mathbf{x}^a_t, \mathbf{x}^b_t$ → Output features $\mathbf{y}_t$

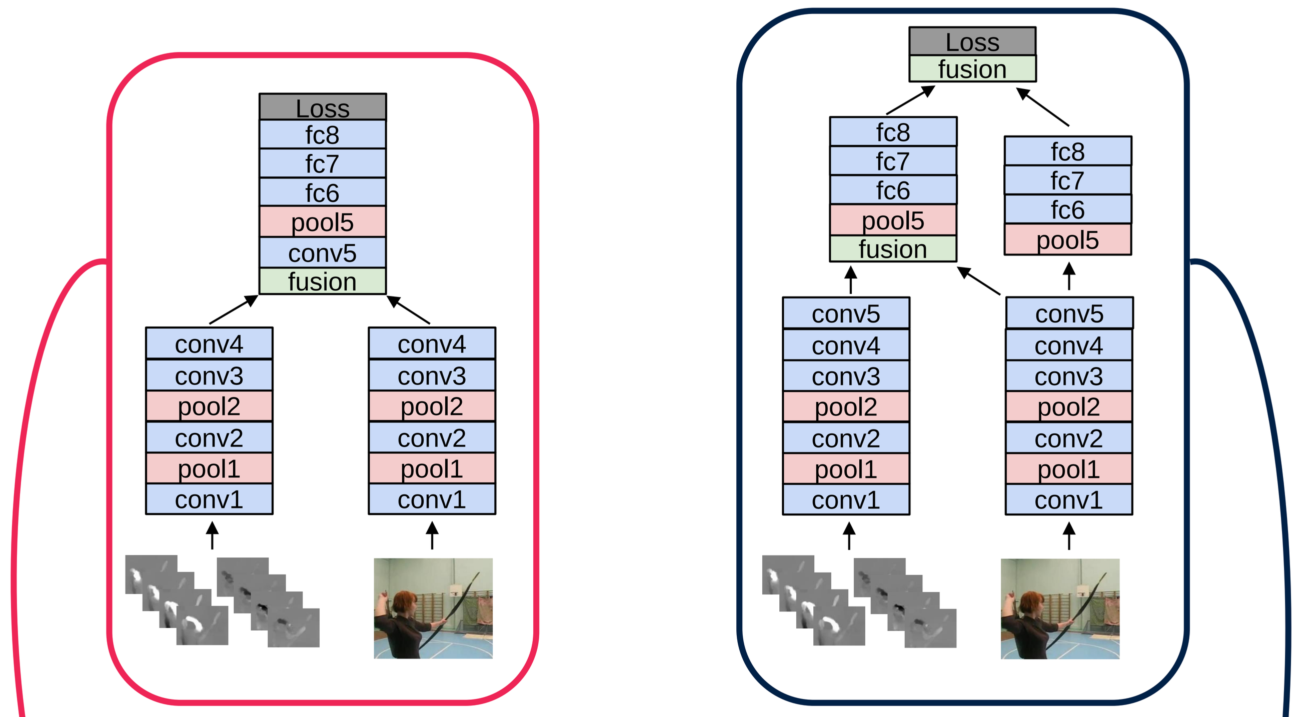
- Sum: $y^{sum}_{i,j,d} = x^a_{i,j,d} + x^b_{i,j,d}$,
- Max: $y^{max}_{i,j,d} = \max\{x^a_{i,j,d}, x^b_{i,j,d}\}$,
- Concatenation $y^{cat}_{i,j,2d} = x^a_{i,j,d}$ $\quad y^{cat}_{i,j,2d-1} = x^b_{i,j,d}$,
- Convolution: $\mathbf{y}^{conv} = \mathbf{y}^{cat} * \mathbf{f} + b$,
- Bilinear: $\mathbf{y}^{bil} = \sum_{i=1}^{H} \sum_{j=1}^{W} \mathbf{x}^{a\top}_{i,j} \otimes \mathbf{x}^{b}_{i,j}$.

Fusion Method	Fusion Layer	Acc.	#layers	#parameters
Sum [1]	Softmax	85.6%	16	181.42M
Sum (ours)	Softmax	85.94%	16	181.42M
Max	ReLU5	82.70%	13	97.31M
Concatenation	ReLU5	83.53%	13	172.81M
Bilinear	ReLU5	85.05%	10	6.61M+SVM
Sum	ReLU5	85.20%	13	97.31M
Conv	ReLU5	85.96%	14	97.58M

Performance on UCF101 (split1)

3. Where to fuse the network streams?

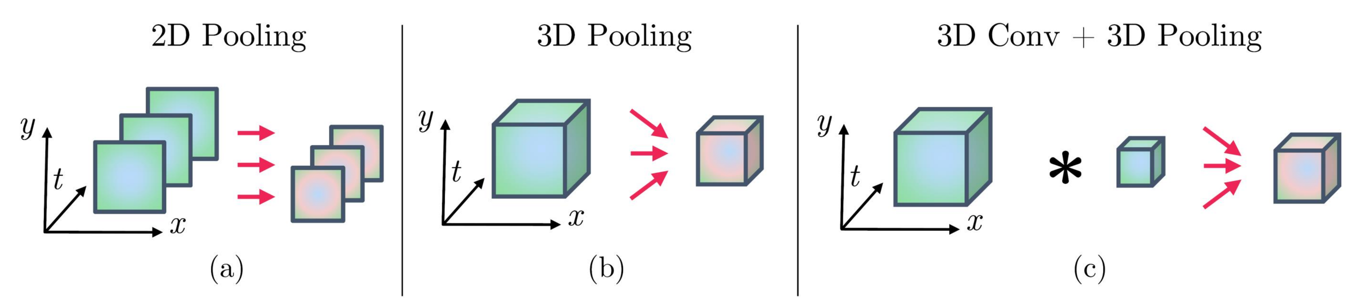
Two examples of where fusion layers can be placed:



Fusion Layers	Accuracy	#layers	#parameters
ReLU2	82.25%	11	91.90M
ReLU3	83.43%	12	93.08M
ReLU4	82.55%	13	95.48M
ReLU5	85.96%	14	97.57M
ReLU5 + FC8	86.04%	17	181,68M
ReLU3 + ReLU5 + FC6	81.55%	17	190,06M

Performance for Conv fusion on UCF101 (split1)

4. How to fuse the two streams temporally?



2D Pooling: Does not combine feature maps over time [1]

3D Pooling: Max-pooling of the temporally stacked features

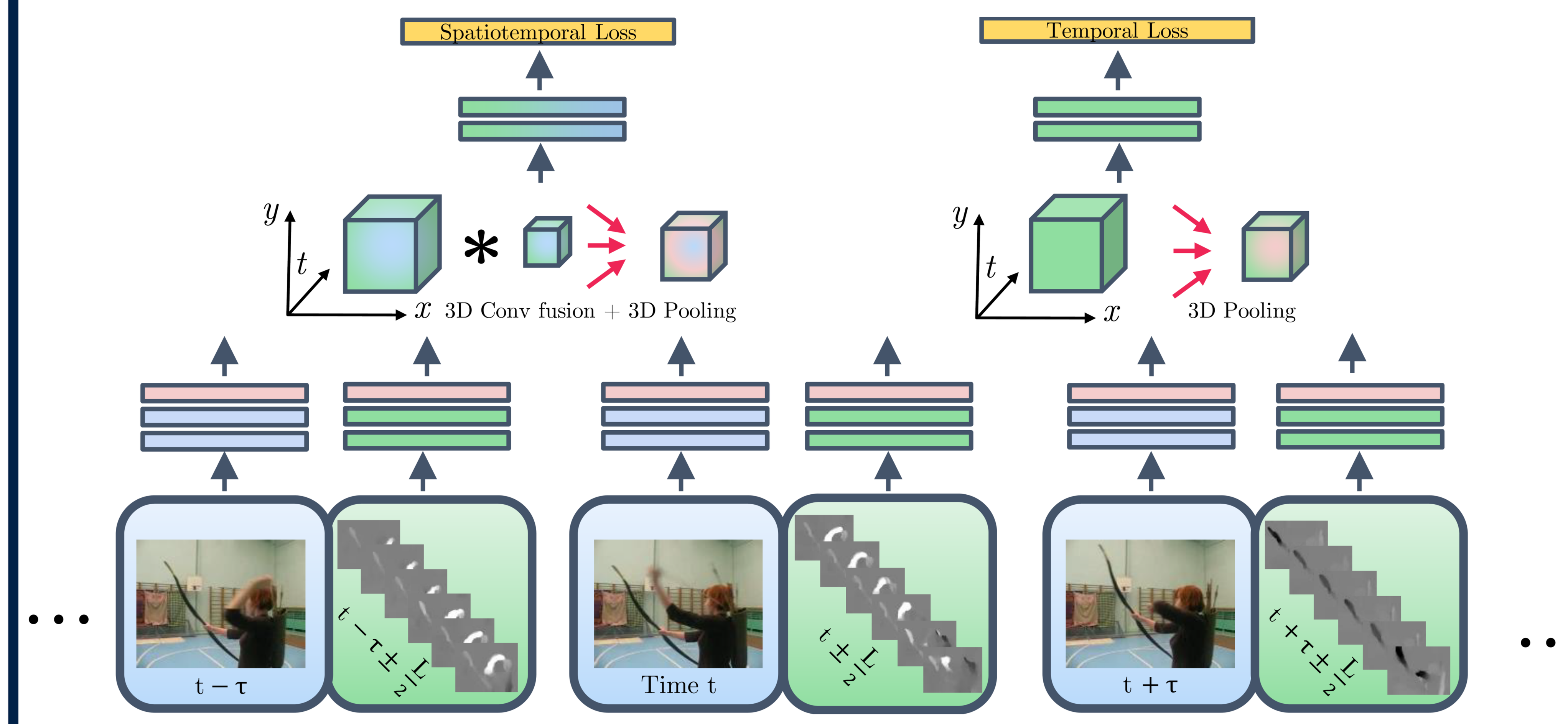
3D Conv + 3D Pooling: Applies convolutional fusion to the temporally stacked features from both streams followed by pooling

Fusion Method	Pooling	Fusion Layers	UCF101	HMDB51
2D Conv	2D	ReLU5 +	89.35%	56.93%
2D Conv	3D	ReLU5 +	89.64%	57.58%
3D Conv	3D	ReLU5 +	90.40%	58.63%

Temporal sampling:

- The temporal fusion layer receives T temporal chunks that are τ frames apart
- This enables us to
  1. Capture short scale ($t \pm \frac{L}{2}$) temporal features at the input of the temporal network (e.g. the drawing of an arrow)
  2. Put them into context over a longer temporal scale ($t + T\tau$) at a higher layer of the network (e.g. drawing an arrow, bending a bow, and shooting an arrow)

5. Proposed Architecture



- Our architecture applies two-stream ConvNets [1] that capture short-term information to temporally adjacent inputs at a coarse temporal scale
- The two streams are fused by a 3D filter that is able to learn correspondences between highly abstract features of the spatial stream and temporal stream
- The resulting features from the fusion stream and the temporal stream are 3D-pooled in space and time to learn spatiotemporal and purely temporal features

6. Comparison with the state-of-the-art

Method	UCF101	HMDB51
C3D [2]	85.2%	-
Two-Stream ConvNet [1]	88.0%	59.4%
Factorized ConvNet [3]	88.1%	59.1%
Two-Stream Conv Pooling [4]	88.2%	-
Ours (S:VGG-16, T:VGG-M)	90.8%	62.1%
Ours (S:VGG-16, T:VGG-16, single tower after fusion)	91.8%	64.6%
Ours (S:VGG-16, T:VGG-16)	92.5%	65.4%
IDT+higher dimensional FV [5]	87.9%	61.1%
C3D+IDT [2]	90.4%	-
TDD+IDT [6]	91.5%	65.9%
Ours+IDT (S:VGG-16, T:VGG-M)	92.5%	67.3%
Ours+IDT (S:VGG-16, T:VGG-16)	93.5%	69.2%

[1] K. Simonyan and A. Zisserman. Two-stream convolutional networks for action recognition in videos. In Proc. NIPS, 2014.
[2] D. Tran, L. Bourdev, R. Fergus, L. Torresani, and M. Paluri. Learning spatiotemporal features with 3D convolutional networks. In Proc. ICCV, 2015.
[3] Lin Sun, Kui Jia, Dit-Yan Yeung, and Bertram Shi. Human action recognition using factorized spatio-temporal convolutional networks. In Proc. ICCV, 2015.
[4] J. Ng, M. Hausknecht, S. Vijayanarasimhan, O. Vinyals, R. Monga, and G. Toderici. Beyond short snippets: Deep networks for video classification. In Proc. CVPR, 2015.
[5] 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, abs/1405.4506, 2014.
[6] Limin Wang, Yu Qiao, and Xiaoou Tang. Action recognition with trajectory-pooled deep-convolutional descriptors. In Proc. CVPR, 2015.

This work was supported by the Austrian Science Fund (FWF) under project P27076, and also by EPSRC Programme Grant Seebibyte EP/M013774/1. The GPUs used for this research were donated by NVIDIA.