

Vision and Perception

Deep Learning for Video: Action Recognition



SAPIENZA
UNIVERSITÀ DI ROMA

References

- Basic reading: Szeliski, Chapter 6.5

Motivation: video classification



track cycling cycling
track cycling
road bicycle racing
marathon
ultramarathon



ultramarathon
ultramarathon
half marathon
running
marathon
inline speed skating



heptathlon
heptathlon
decaathlon
hurdles
pentathlon sprint
(running)



bikejoring
mushing
bikejoring
harness racing
skijoring
carting



longboarding
longboarding aggressive
inline skating freestyle
scootering freeboard
(skateboard) sandboarding



ultimate (sport)
ultimate (sport)
hurling flag football
association football
rugby sevens

Activity classification is the task of identifying a pre-defined set of physical actions (gestures, actions, human-object interactions, interactions, group activities)

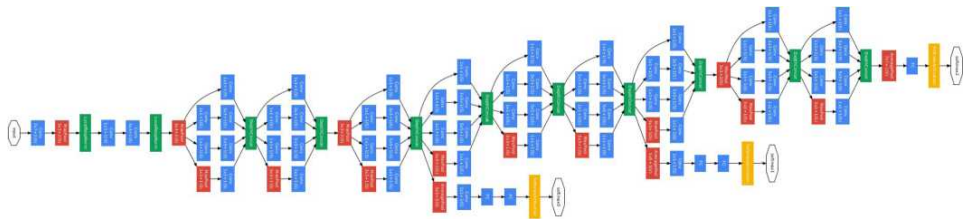
What is a video?

- Formally, a video is a 3D signal
 - o Spatial coordinates: x, y
 - o Temporal coordinate: t
- If we fix t , we obtain an image. We can understand videos as sequences of images (a.k.a. frames)



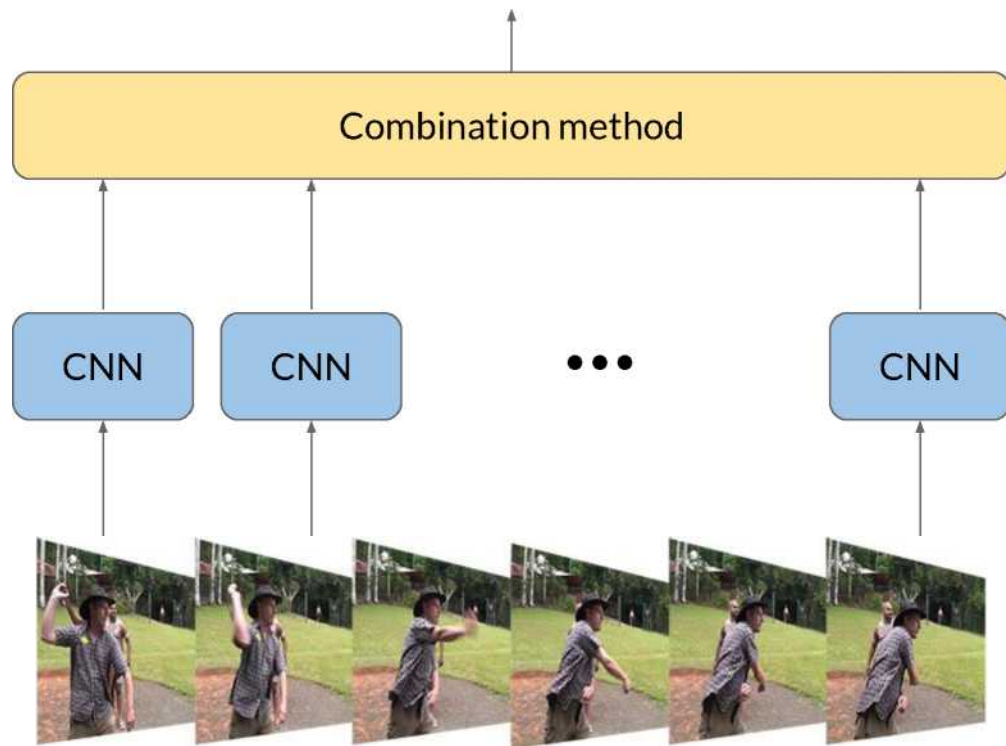
How do we work with images?

Convolutional Neural Networks (CNN) provide state of the art performance on image analysis tasks

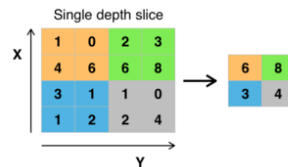


How do we work with videos? How can we extend CNNs to image sequences?

Single frame models



Combination is commonly implemented as a small NN on top of a pooling operation (e.g. max, sum, average).



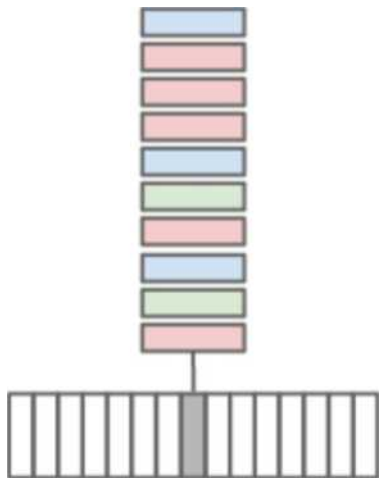
Pro: to reduce the resolution of the feature map retaining features required for classification

Cons: pooling is not aware of the temporal order!

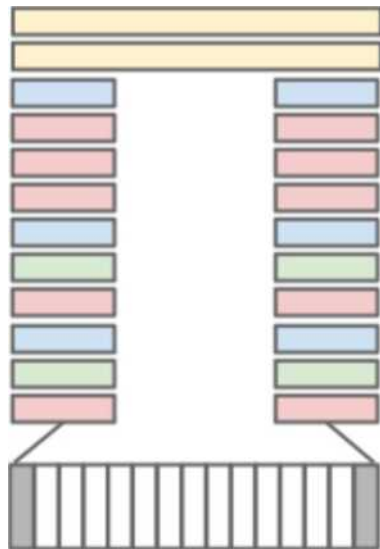
Multiple Frames

Explored approaches for fusing information over temporal dimension through the network

Single Frame

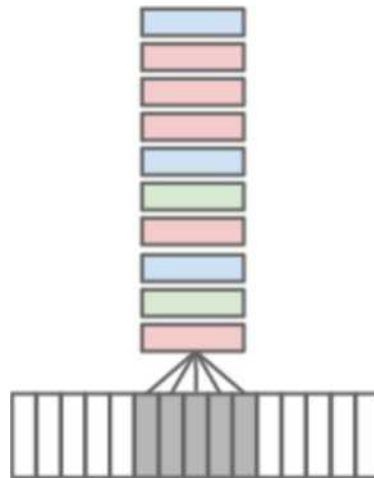


Late Fusion



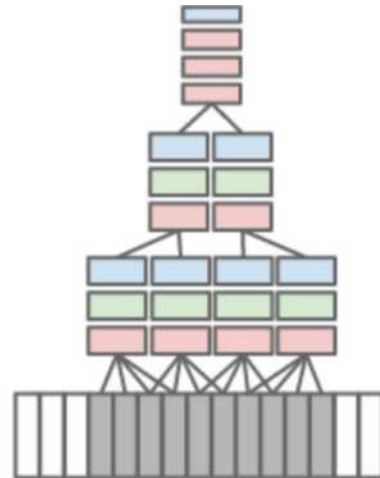
place two separate single-frame networks up to last conv layers with shared parameters a distance of 15 frames apart and then merge the two streams in the first fully connected layers

Early Fusion



11x11x3xT filters on the first convolutional layer
T=10 frames

Slow Fusion



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

Multiple Frames

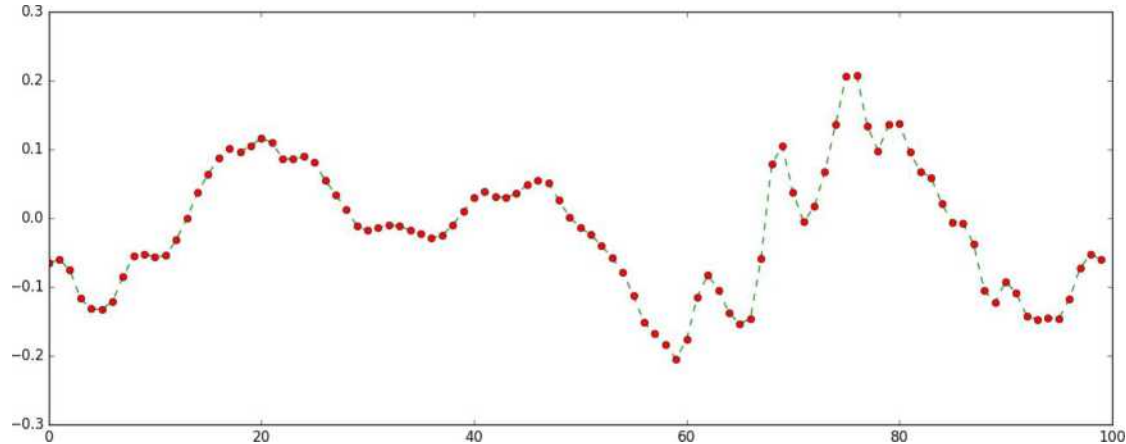
Model	Clip Hit@1	Video Hit@1	Video Hit @5
Feature Histograms + Neural Net	.	55.3	.
Single-Frame	41.1	59.3	77.7
Early Fusion	38.9	57.7	76.8
Late Fusion	40.7	59.3	78.7
Slow Fusion	41.9	60.9	80.2
CNN Average (Single+Early+Late+Slow)	41.4	63.9	82.4

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

Limitation of Feed Forward NN (as CNNs)

If we have a sequence of samples...

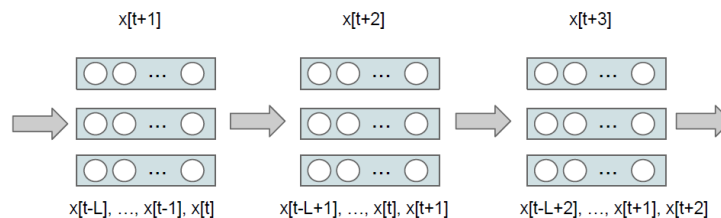


predict sample $x[t+1]$ knowing previous values $\{x[t], x[t-1], x[t-2], \dots, x[t-T]\}$

Limitation of CNNs

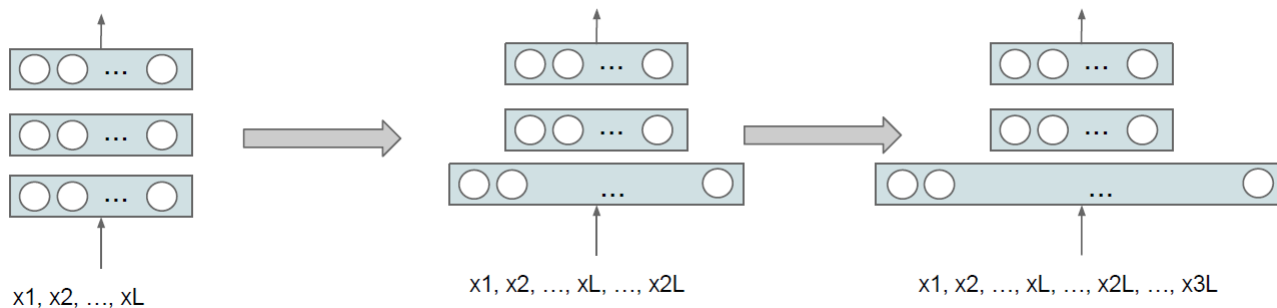
Feed Forward approach:

- static window of size L
- slide the window time-step wise

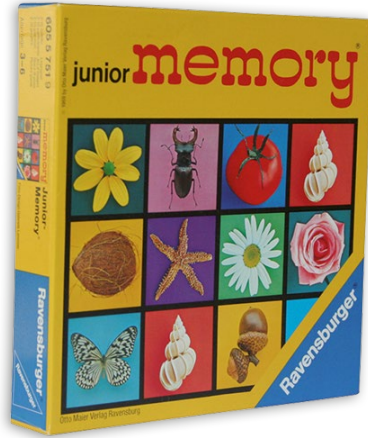


Problems for the feed forward + static window approach:

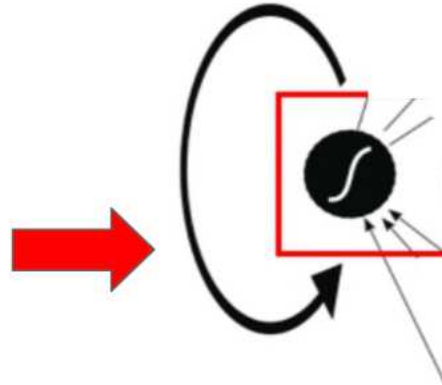
- What's the matter increasing L ? \rightarrow Fast growth of num of parameters
- Decisions are independent between time-steps
 - The network doesn't care about what happened at previous time-step, only present window matters
- Cumbersome padding when there are not enough samples to fill L size
 - Can't work with variable sequence lengths



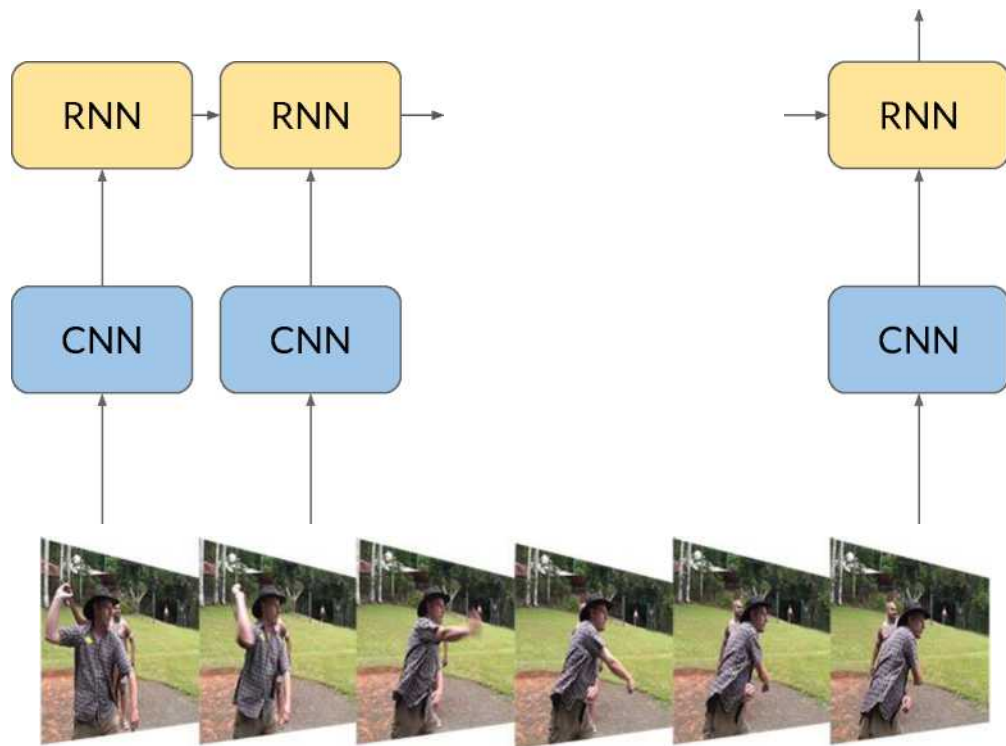
Recurrent Neural Network (RNN)



The hidden layers and the output depend from previous states of the hidden layers



2DCNN + RNN

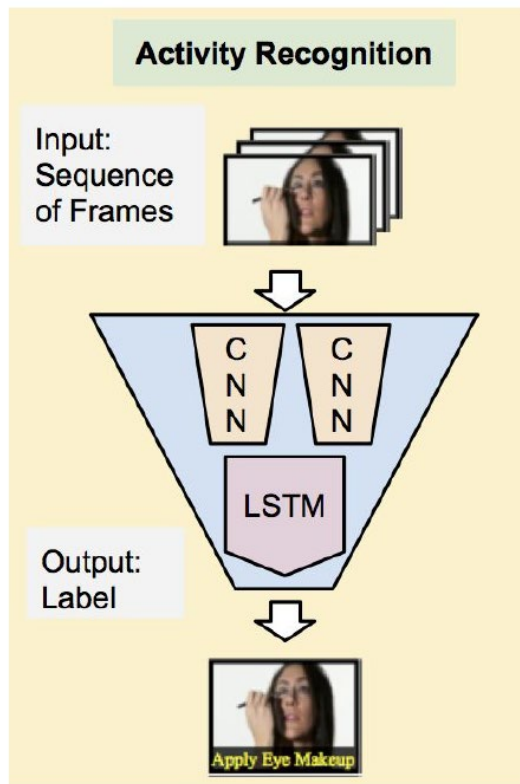


Pro: Recurrent Neural Networks are well suited for processing sequences.

- Used to model sequences of per-frame CNN representations

Cons: RNNs are sequential and cannot be parallelized.

2D CNN + RNN



Jeffrey Donahue, Lisa Anne Hendricks, Sergio Guadarrama, Marcus Rohrbach, Subhashini Venugopalan, Kate Saenko, Trevor Darrel. [Long-term Recurrent Convolutional Networks for Visual Recognition and Description](#), CVPR 2015.

2D CNN + RNN



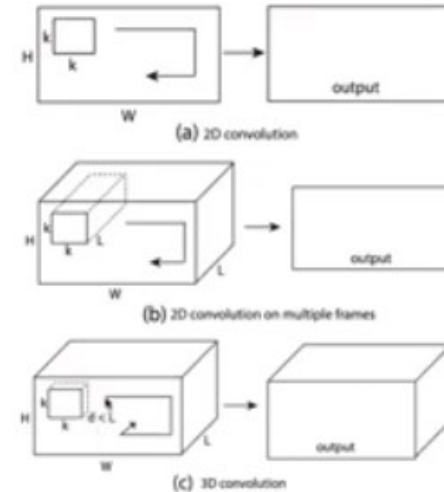
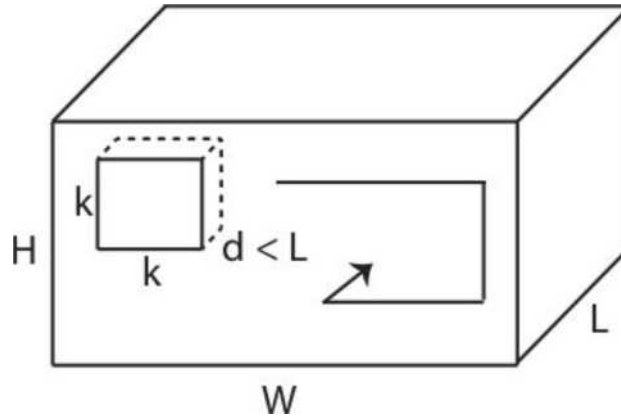
Used Unused

Victor Campos, Brendan Jou, Xavier Giro-i-Nieto, Jordi Torres, and Shih-Fu Chang. [“Skip RNN: Learning to Skip State Updates in Recurrent Neural Networks”](#), ICLR 2018.

3D CNN (C3D)

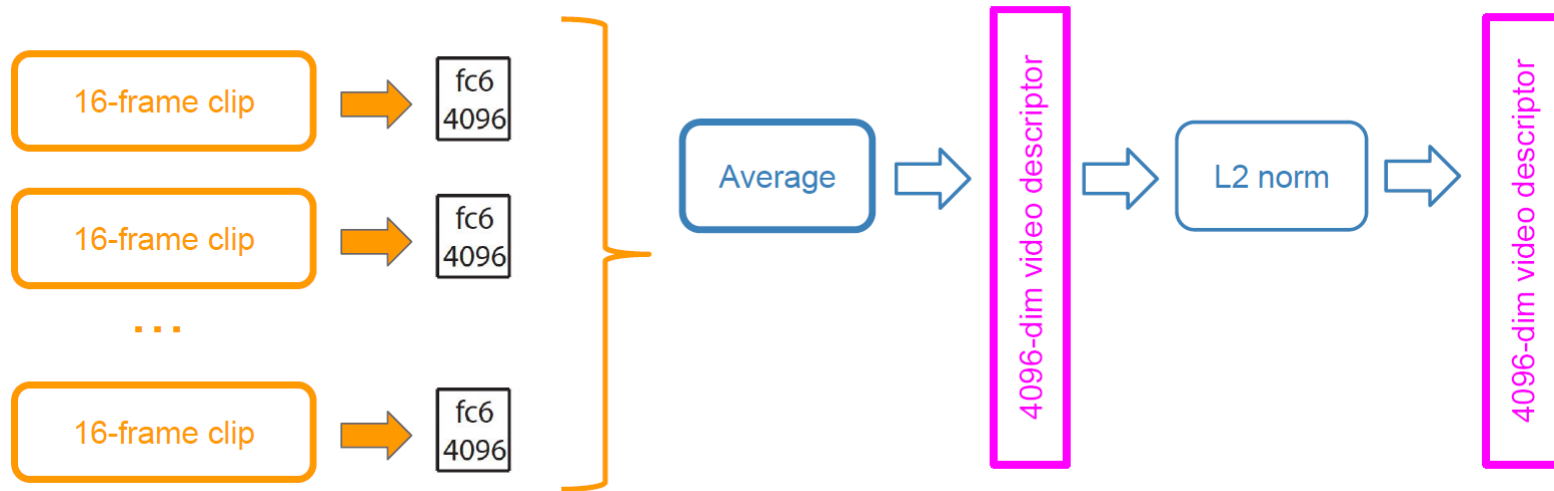
We can add an extra dimension to standard CNNs:

- An image is a $H \times W \times D$ tensor: $M \times N \times D'$ conv filters
- A video is a $L \times H \times W \times D$ tensor: $K \times M \times N \times D'$ conv filters



3D CNN (C3D)

The video needs to be split into chunks (also known as *clips*) with a number of frames that fits the receptive field of the C3D. Usually clips have 16 frames.



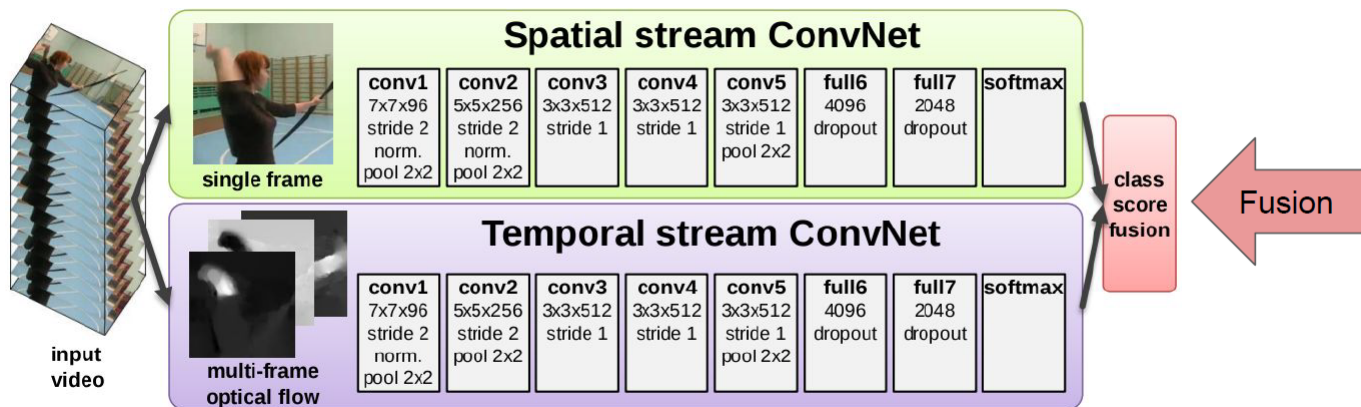
Limitations:

- How can we handle longer videos?
- How can we capture longer temporal dependencies?

Two-streams 2D CNNs

Problem: Single frame models do not take into account motion in videos.

Solution: extract optical flow for a stack of frames and use it as an input to a CNN.



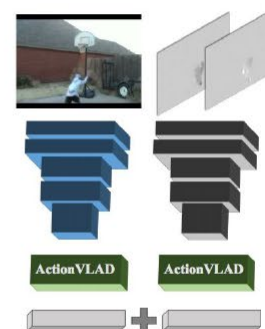
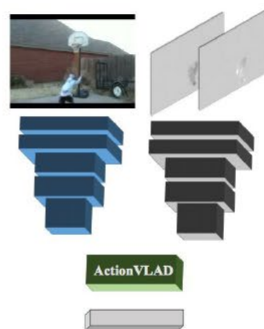
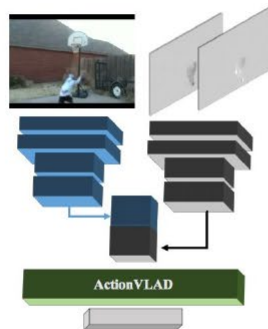
Two-streams 2D CNNs

Vector of Locally Aggregated Descriptor

Effect of
training
(rgb/flow)

Two-Stream	VLAD	ActionVLAD
47.1/55.2	44.9/55.6	51.2/58.4

How to
fuse RGB
and Flow?



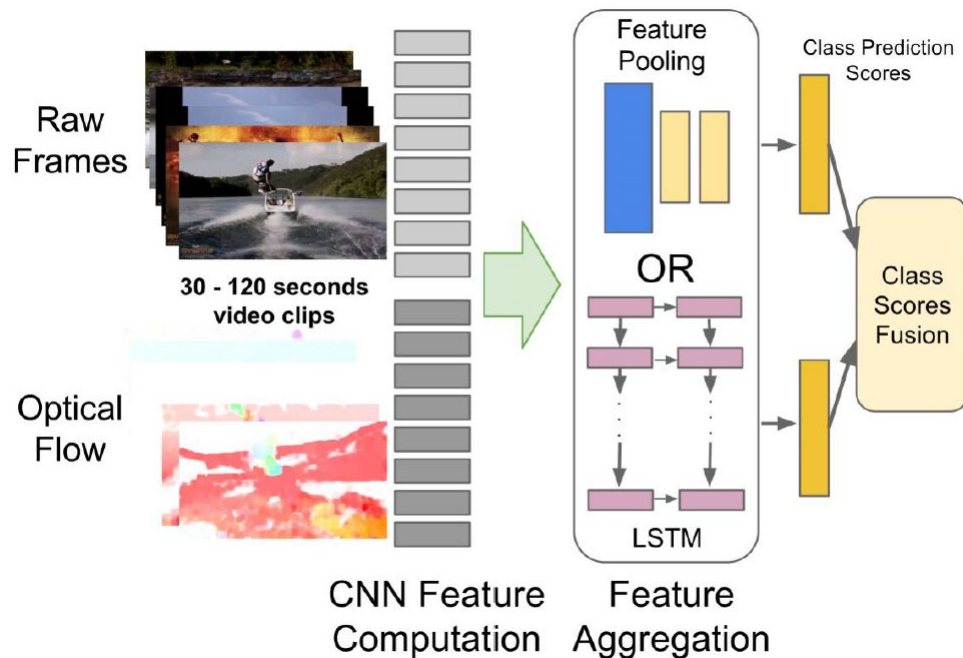
ActionVLAD
pooling layer

Concat Fuse	Early Fuse	Late Fuse
56.0	64.8	66.9

<https://github.com/rohitgirdhar/ActionVLAD>

Girdhar, Rohit, Deva Ramanan, Abhinav Gupta, Josef Sivic, and Bryan Russell. "ActionVLAD: Learning spatio-temporal aggregation for action classification." CVPR 2017.

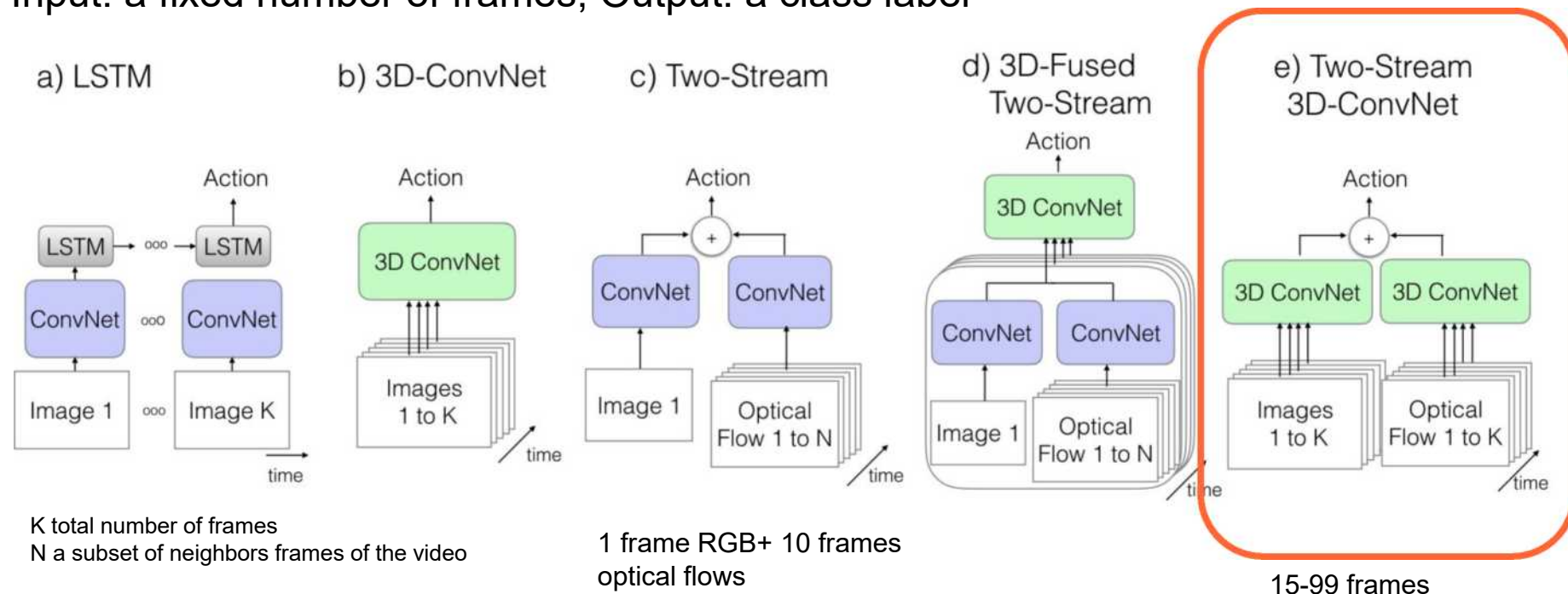
Two-streams 2D CNNs + RNN



Yue-Hei Ng, Joe, Matthew Hausknecht, Sudheendra Vijayanarasimhan, Oriol Vinyals, Rajat Monga, and George Toderici. "Beyond short snippets: Deep networks for video classification." CVPR 2015

Two-streams 3D CNNs

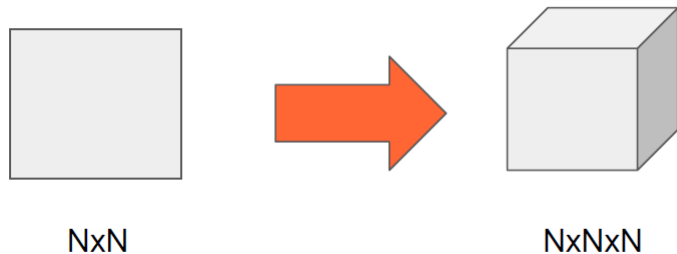
Input: a fixed number of frames, Output: a class label



Two-streams Inflated 3D CNNs (I3D)

Adapt 2D CNNs found for ImageNet classification to 3D convolutions

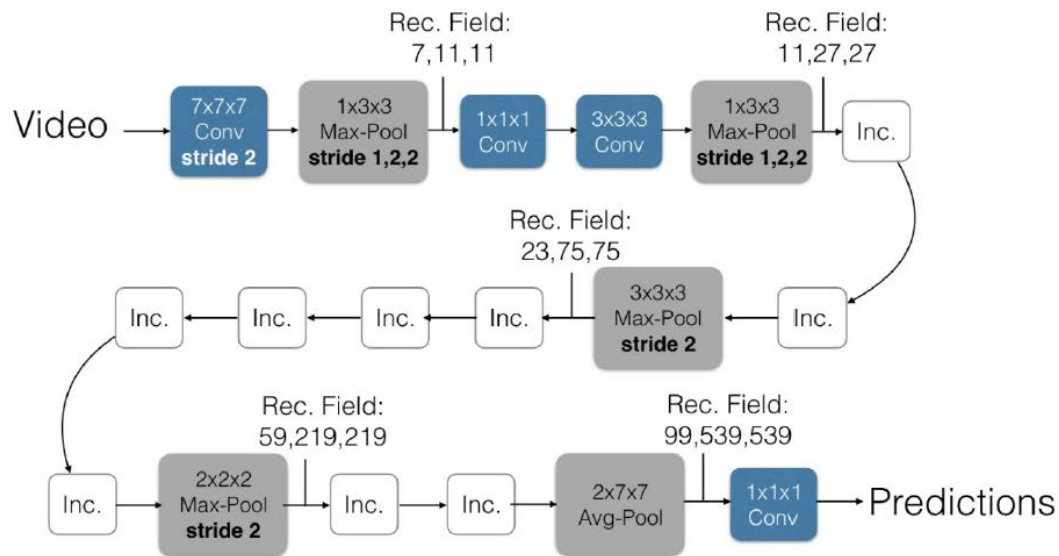
Repeating the weights of the 2D filters N times along the time dimension and rescaling them by dividing by N



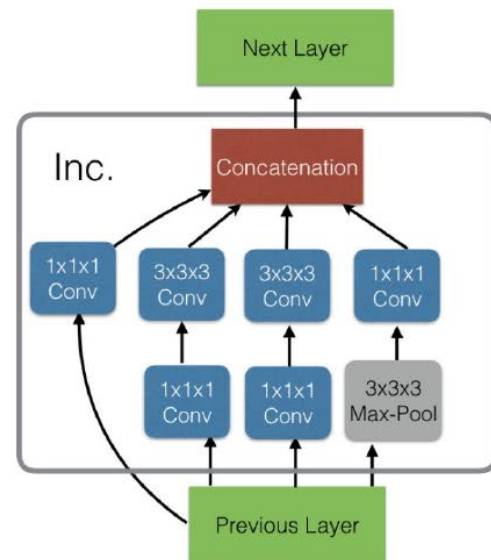
- 3D models are initialized with ImageNet images transformed into 'boring' video sequences.
- Train the two networks separately and average their prediction at test time
- Inception architecture

Two-streams 3D CNNs

Inflated Inception-V1



Inception Module (Inc.)



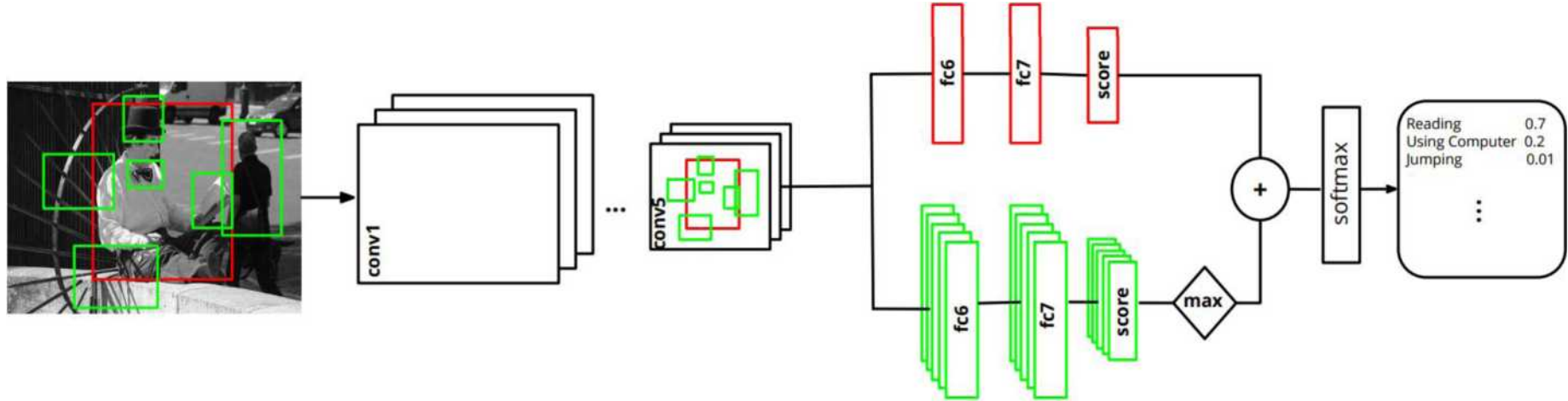
CNNs for sequences of images

CNN Input	RGB	Optical Flow	Fusion
Single frame	2D CNN	.	Pooling + NN
Multiple frames	2D CNN	.	Pooling + NN
Sequence of images	2D CNN	.	RNN
Sequence of clips	3D CNN	.	Pooling
Sequence of clips	3D CNN	.	RNN
Two-stream	2D CNN	2D CNN	Pooling
Two-stream	2D CNN	2D CNN	RNN
Two-stream	Inflated 3D CNN	Inflated 3D CNN	Pooling

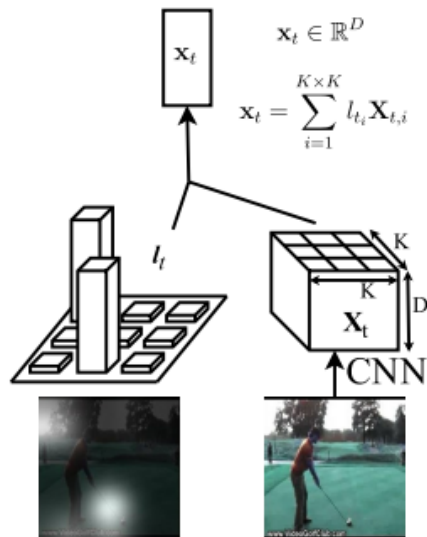
Action recognition

Which deep learning techniques at a local scale may help action recognition ?

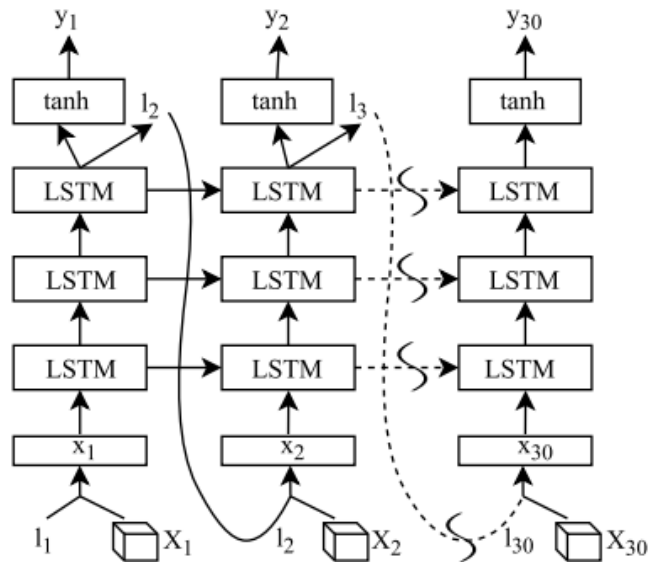
Action Recognition with object detection



Action Recognition with attention



(a) The soft attention mechanism

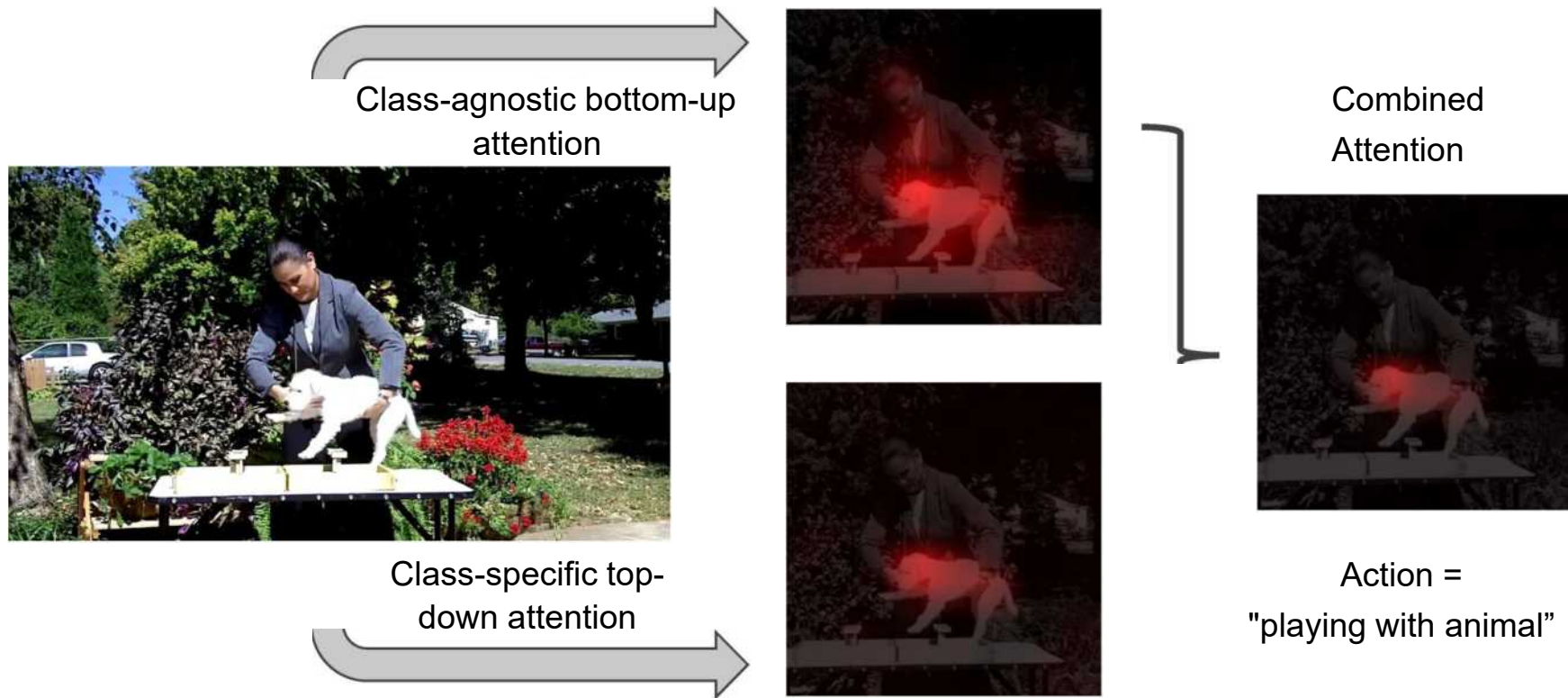


(b) Our recurrent model

Action Recognition with soft attention



Action recognition with soft attention



Action Recognition with hard attention



Figure 1. Key volumes detected by our key volume mining deep framework. A volume is a spatial-temporal video clip. The top row shows key volumes are very sparse among the whole video, and the second row shows that key volumes may come from different modalities (different motion patterns here). Note that frames are sampled with fixed time interval.

Soft vs Hard Attention model

Hard attention:

- Attent to a single input location.
- Can't use gradient descent.
- Need **reinforcement learning**.

Soft attention:

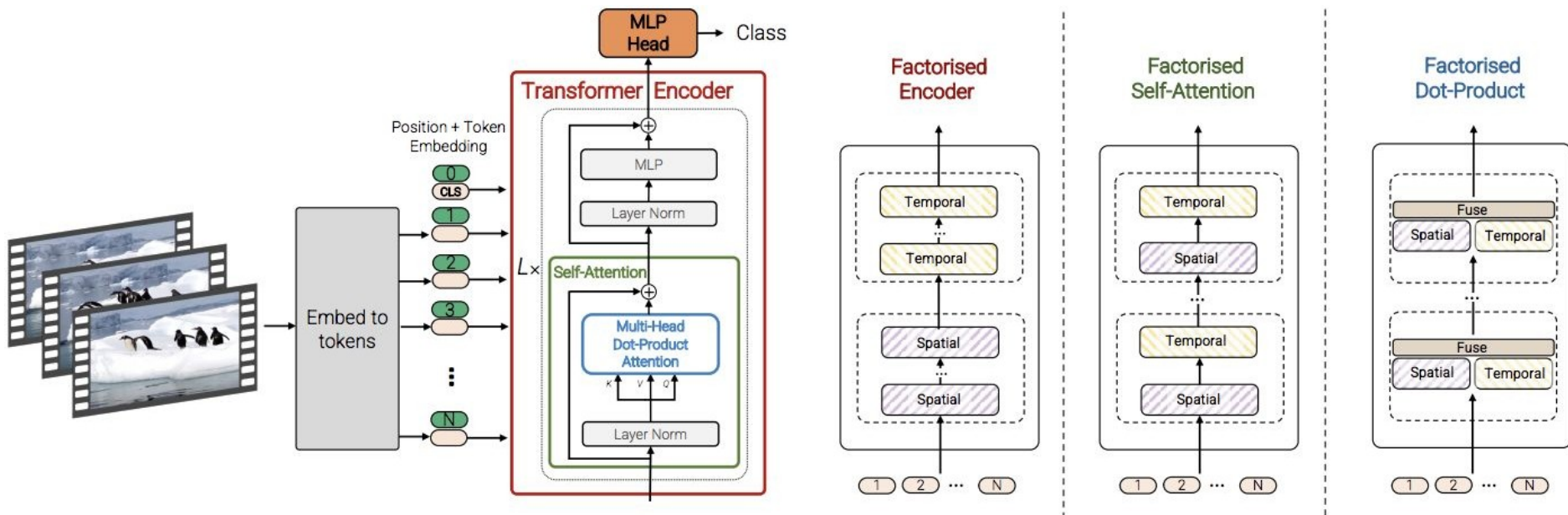
- Compute a weighted combination (attention) over some inputs using an attention network.
- Can use backpropagation to train end-to-end.

Self-attention

- **Self-attention**, a highly **efficient** technique that makes it possible to update the embeddings of every word in the input text in **parallel**.
- **Self-attention**, also known as **intra-attention**, is an attention mechanism relating different positions of a single sequence in order to compute a representation of the same sequence.
 - It has been shown to be very useful in machine reading, abstractive summarization, or image description generation.

Transformers for Video Recognition

Video ViT



Video ViT

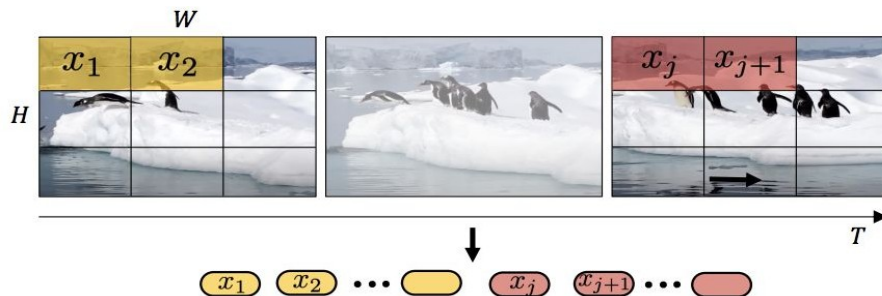


Figure 2: Uniform frame sampling: We simply sample n_t frames, and embed each 2D frame independently following ViT [15].

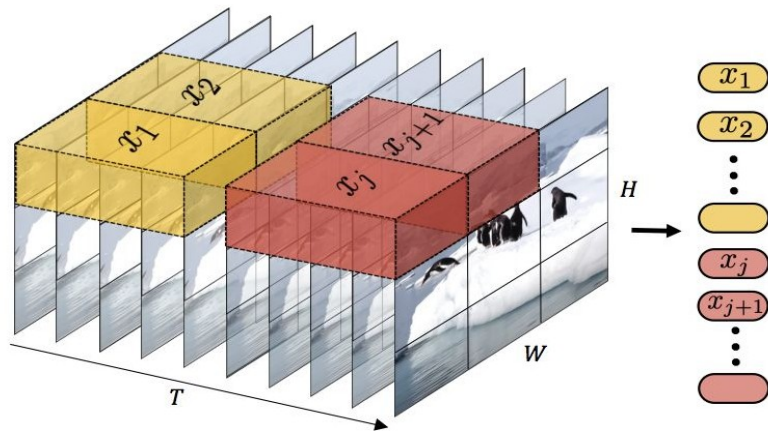
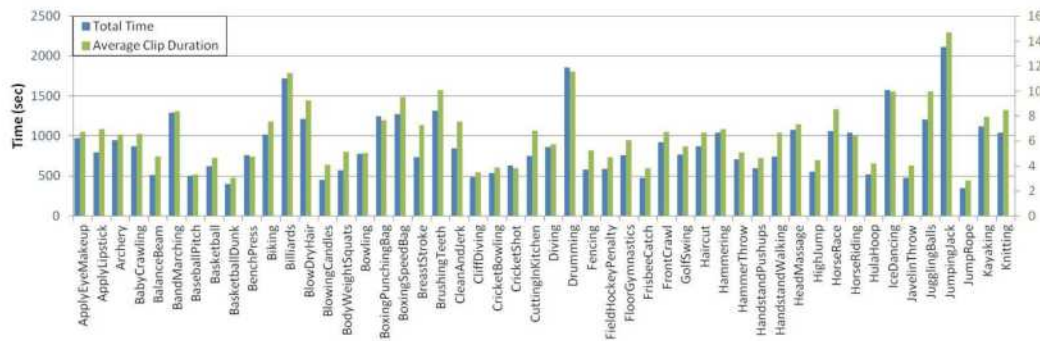


Figure 3: Tubelet embedding. We extract and linearly embed non overlapping tubelets that span the spatio-temporal input volume

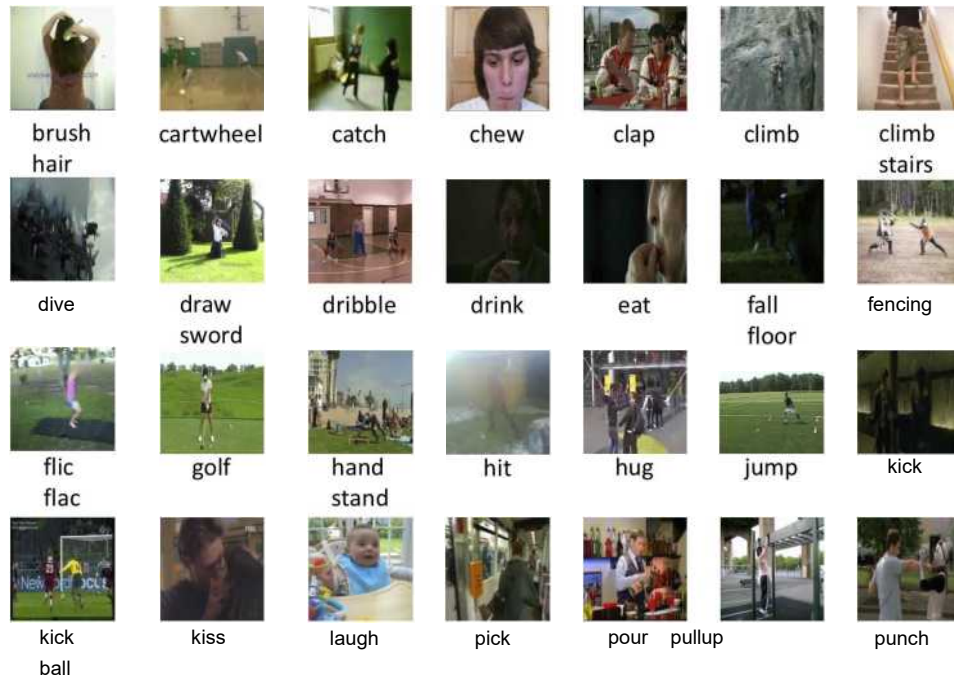
Datasets: UCF-101



Youtube, 101 categories

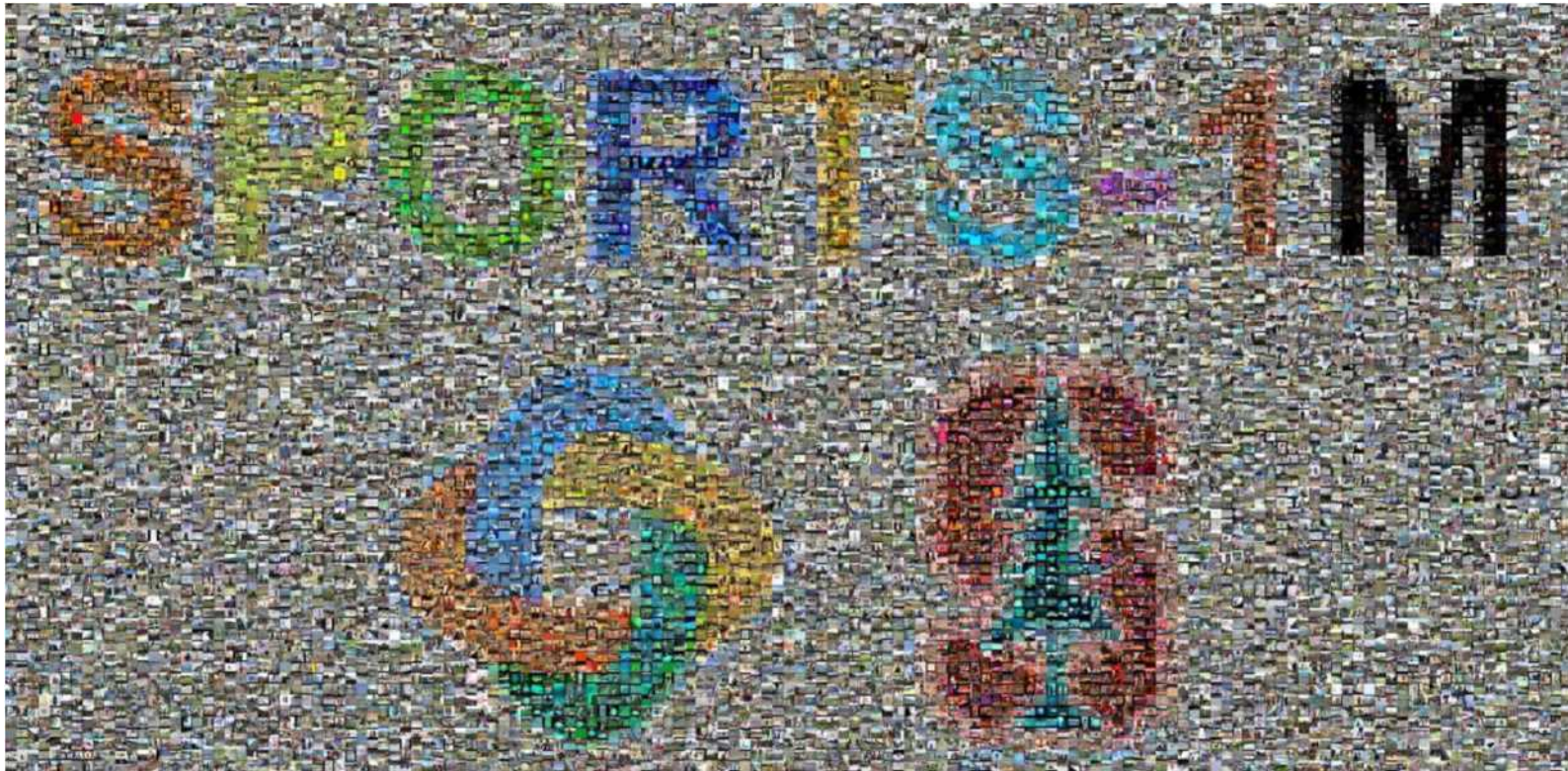
Soomro, K., Zamir, A. R., & Shah, M. (2012). UCF101: A dataset of 101 human actions classes from videos in the wild. arXiv preprint arXiv: 1212.0402.

Datasets: HMDB51 (Brown University)



Kuehne, Hildegard, Hueihan Jhuang, Estfbaliz Garrote, Tomaso Poggio, and Thomas Serre. "HMDB: a large video database for human motion recognition." ICCV 2011.

Datasets: Sports 1M (Stanford)



487 classes of sports

Karpathy, A., Toderici, G., Shetty, S., Leung, T., Sukthankar, R., & Fei-Fei, L. Large-scale video classification with convolutional neural networks. CVPR2014.

Datasets: Kinectics (DeepMind)

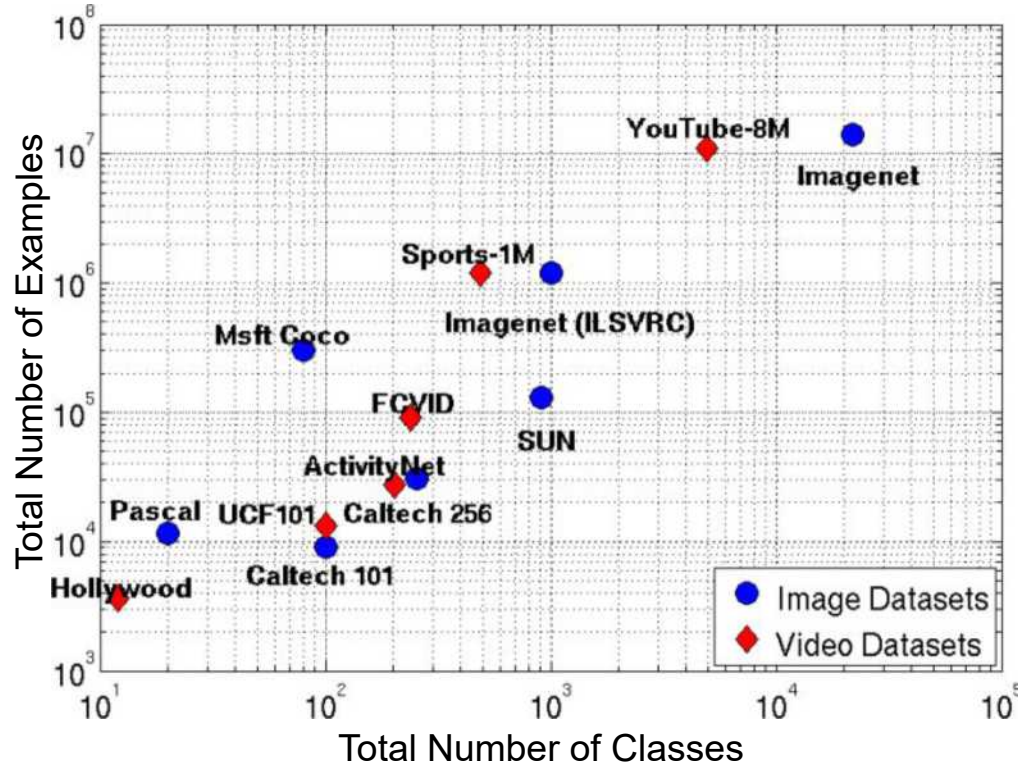
DEEPMIND
KINETICS VIEWER



400 human action classes – 400 or more clips per class

Kay, W., Carreira, J., Simonyan, K., Zhang, B., Hillier, C., Vijayanarasimhan, S., ... & Suleyman, M. (2017). [The kinetics human action video dataset](#). arXiv preprint arXiv:1705.06950.

Activity Recognition: Datasets



Large-scale datasets

- The reference dataset for image classification, ImageNet, has ~1.3M images
 - Training a state of the art CNN can take up to 2 weeks on a single GPU
- Now imagine that we have an 'ImageNet' of 1.3M videos
 - Assuming videos of 30s at 24fps, we have 936M frames
 - This is 720x ImageNet!
- Videos exhibit a large redundancy in time
 - We can reduce the frame rate without losing too much information

Memory issues

- Current GPUs can fit batches of 32~64 images when training state of the art CNNs
 - This means 32~64 video frames at once
- Memory footprint can be reduced in different ways if a pre-trained CNN model is used
 - Freezing some of the lower layers, reducing the memory impact of backprop
 - Extracting frame-level features and training a model on top of it (e.g. RNN on top of CNN features). This is equivalent to freezing the whole architecture, but the CNN part needs to be computed only once.

I/O bottleneck

- In practice, applying deep learning to video analysis requires for multi-GPU or distributed settings
- In such settings it is very important to avoid *starving* the GPUs or we will not obtain any speedup
 - The next batch needs to be loaded and preprocessed to keep the GPU as busy as possible
 - Using asynchronous data loading pipelines is a key factor
 - Loading individual files is slow due to the introduced overhead, so using other formats such as TFRecord/HDF5/LMDB is highly recommended

Acknowledgement: some slides and material from Bernt Schiele, Mario Fritz, Michael Black, Bill Freeman, Fei-Fei, Justin Johnson, Serena Yeung, R. Szelisky, Fabio Galasso, Ioannis Gkioulekas, Noah Snavely, Abe Davis, Kris Kitani, Xavier Giró-i-Nieto