Video

Computer Vision – Lecture 12

Further Reading

- Slides from L Fei-Fei
- Slides from J Johnson
- Slides from M Niessner & L Leal-Taixé (2nd part)

Videos

- Sequence of frames: T x 3 x H x W
- Frame rate: ~30 FPS (frames per second)
- Temporal coherence
- Sometimes: shot changes/skips

Video source

Video Tasks

- Activity Classification
- Temporal/Spatial Action Localisation
- Event Dense Captioning
- Active Speaker Recognition
- Sign Language Transcription
- …

Videos

Problem: videos are big!

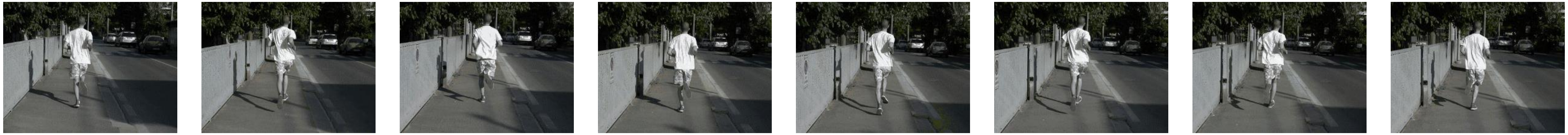
- HD (1080x1920):
  - 60*30*3*1080*1920 bytes = 11.2GB / minute

- This is just the data, we still need to compute things.

- Use short, low-res clips: T=16, FPS=5, H=W=112: (60MB/min)

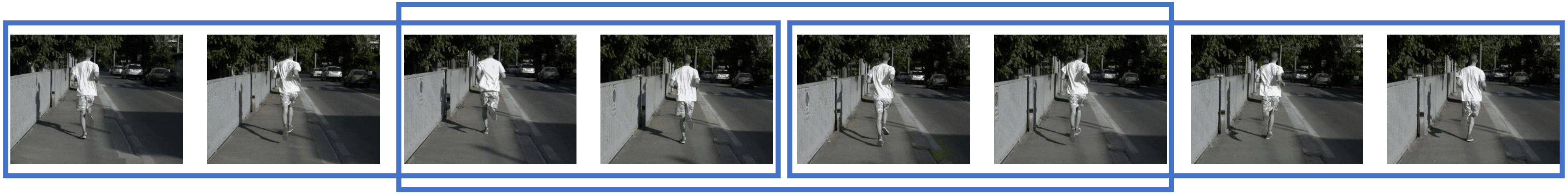
Windowed Video Processing

Raw Video: long, high resolution, high FPS



Training: short clips, low resolution, low FPS

Testing: run model on overlapping clips, average predictions



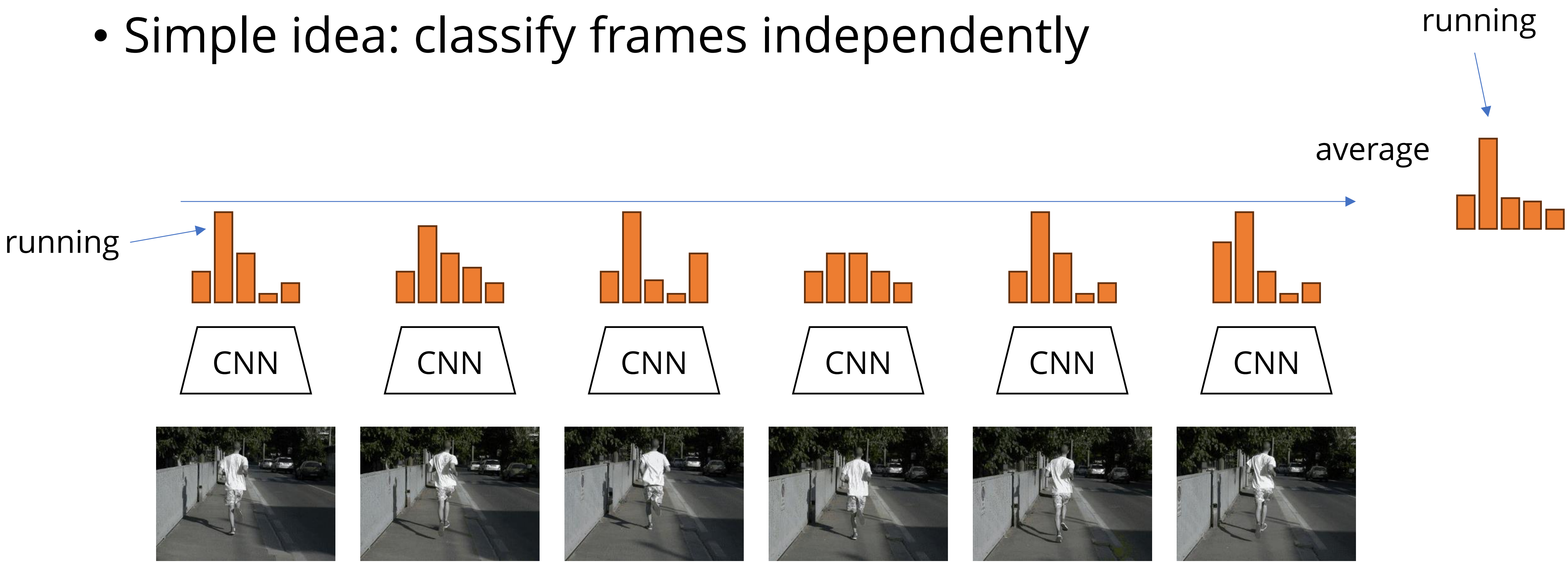
Windowed Video Processing

Familiar ideas:

- Subsampling (now in time and space)
- Sliding window (now mostly in time)
- Share as much computation as possible to increase the efficiency

Task: Video Classification

- Action recognition: running, knitting, basketball, etc.
- Simple idea: classify frames independently



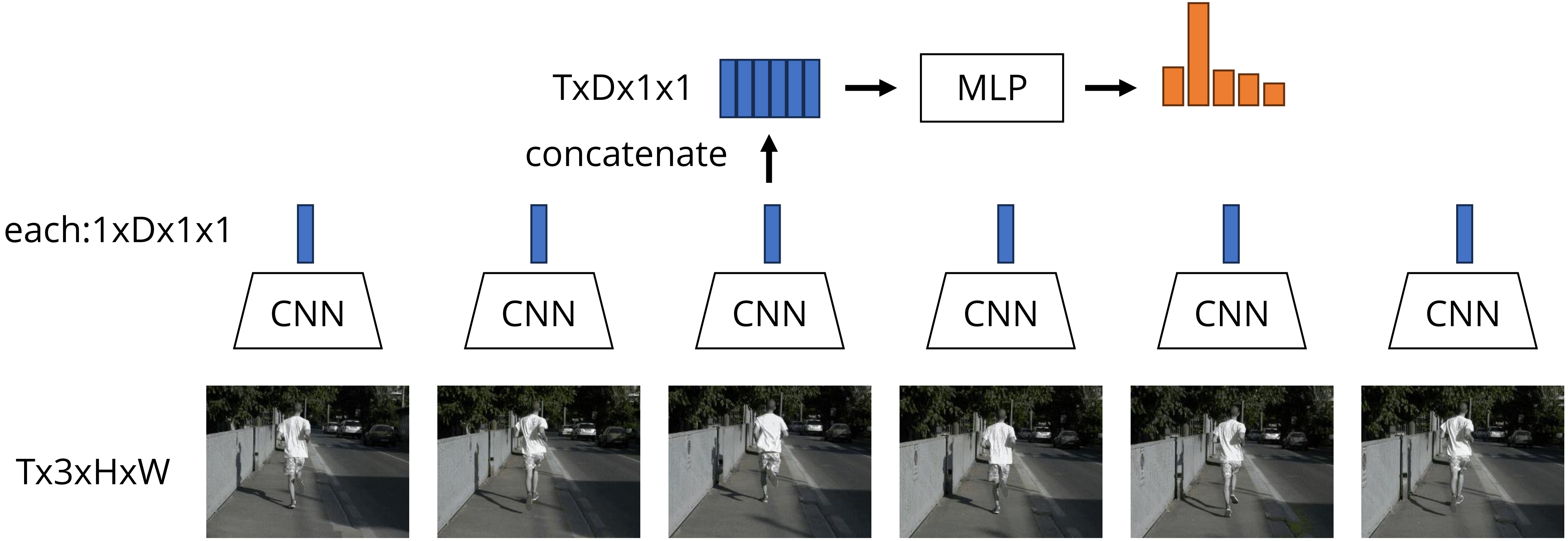
Per-Frame Models

- Predict for each frame independently with an image model.
- Average probabilities (mathematically questionable but works much better than multiplication).

- Often a very strong baseline!
- Intuition:
  - Only one frame is needed to differentiate between “running” and “swimming”.
  - This is often called: object-bias of action recognition.
- Depends on task: “sitting down” vs. “standing up” needs motion.

Sharing Computation

- Per-frame model cannot reason about time: we only average predictions.
- Add layers that have access across time.



Late Fusion

- Here we decided to include temporal information “late” in the computation.
- Intuition: get per-frame high-level understanding, then combine them across time.

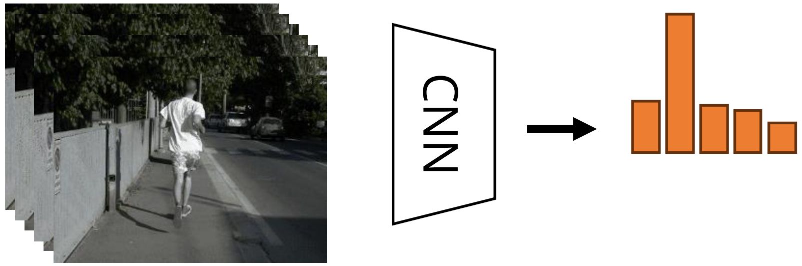
- Fusion mechanisms: concatenation, pooling, etc.

- Problem: low-level motion often lost after “compressing” a frame into a feature vector.

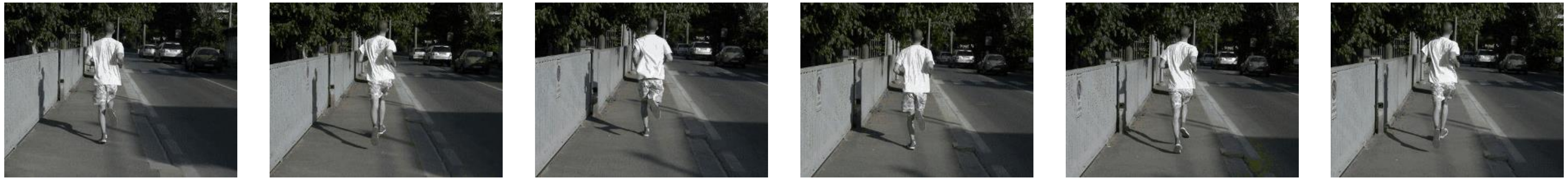
Early Fusion

- Fuse frames at the input level: Tx3xHxW -> T3xHxW
- Treat input as an image with many channels.

T3xHxW



Tx3xHxW



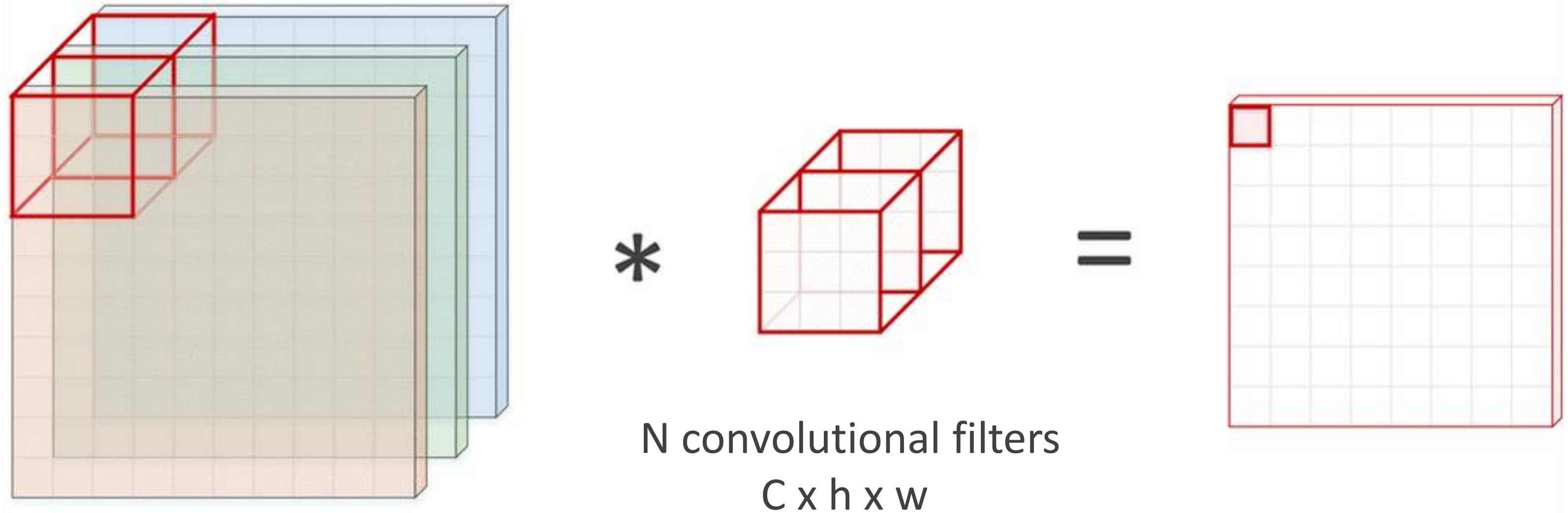
Early Fusion

- First layer takes as input the whole video stacked in the channel dimension.

- Problem: effectively only the first layer has access to temporal information
- First 2D convolution collapses all temporal information: 3TxHxW -> DxH’xW’

- Remaining network is a standard 2D network.

2D Convolution

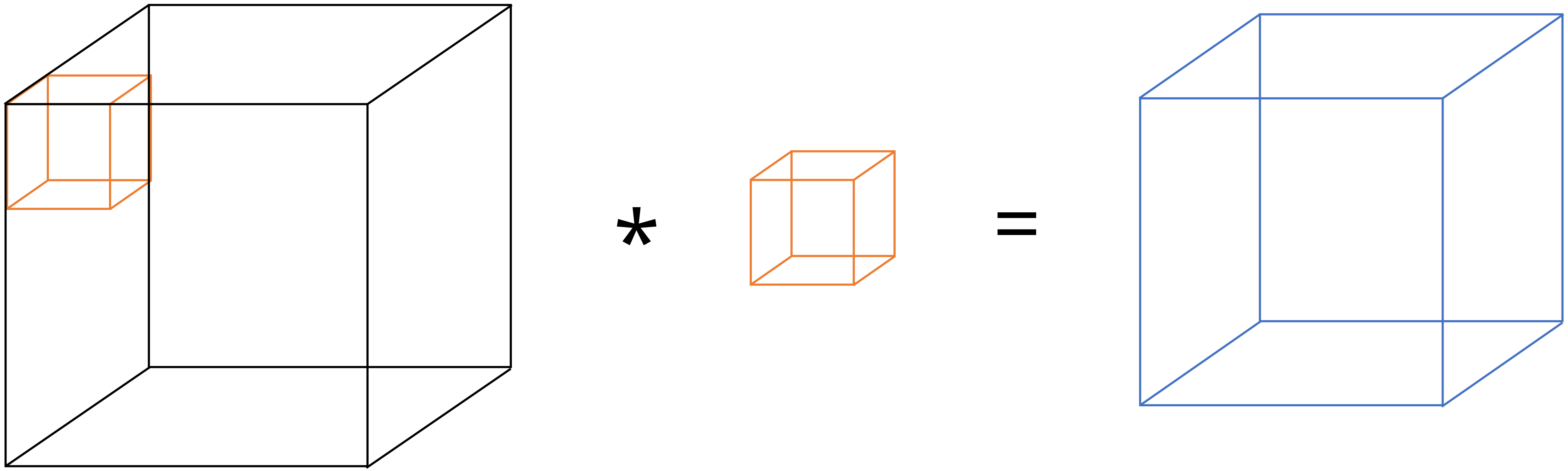


N convolutional filters
C x h x w

Input
C x H x W

Output
N x H x W
(with appropriate padding)

3D Convolution



Input
C x D x H x W

N Filters
C x d x h x w

Output
N x D x H x W

Not shown: channel dimension

3D Convolution

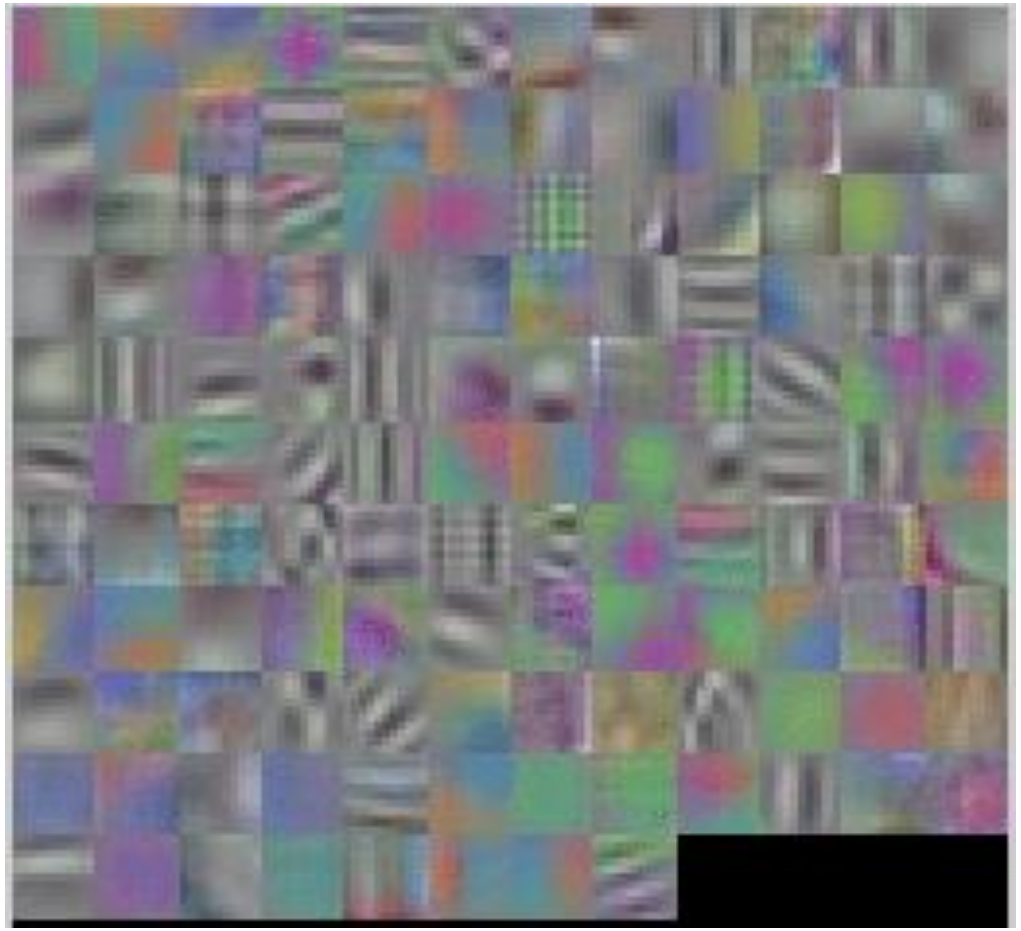
- The convolution now slides along 3 directions: width, height, and depth.
- There is still a channel dimension: each feature in the 3D feature map has C dimensions.
- The output is also a 3D feature map.
- Naming: we ignore the channel dimension. Input and output are actually 4D tensors. (weights: 5D)

Other operations

- 3D Pooling: works the same way. Filter size e.g. 2x2x2
- Activation: works element-wise. (no change)
- Fully connected layer: same as in 2D. Reshape into a vector before applying it. (or use a convolution that has the same size as the feature map)
- 3D CNN: swap 2D operations with 3D operations.

First Layer Filters

- Filters span space and time.
- We can visualise them by animating them through time.
- Moving edge filters.
- Not all filters change with time.



Large-scale Video Classification with Convolutional Neural Networks, Karpathy et al., 2014

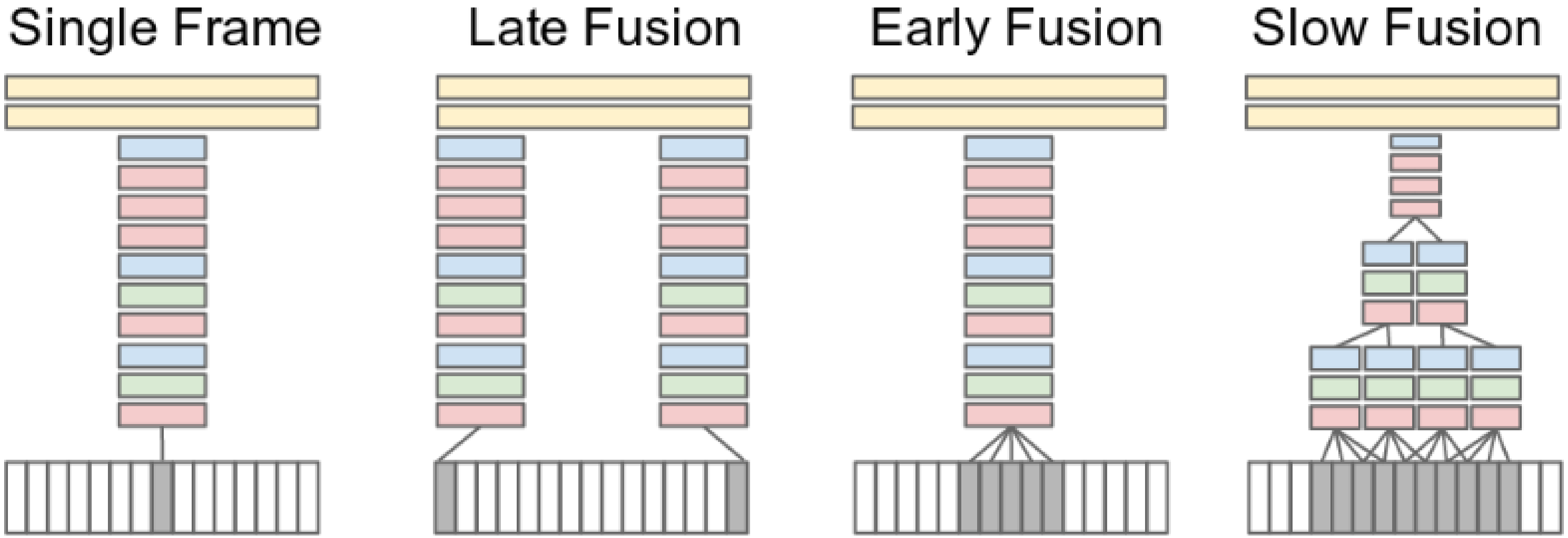
3D CNNs for Video Understanding

- Slow Fusion: slowly fuse temporal information over the course of the network

- Adds shift invariance in time (same motion at a different time).

- Subsampling in space and time gives larger and larger context to each successive layer.

Fusion Approaches

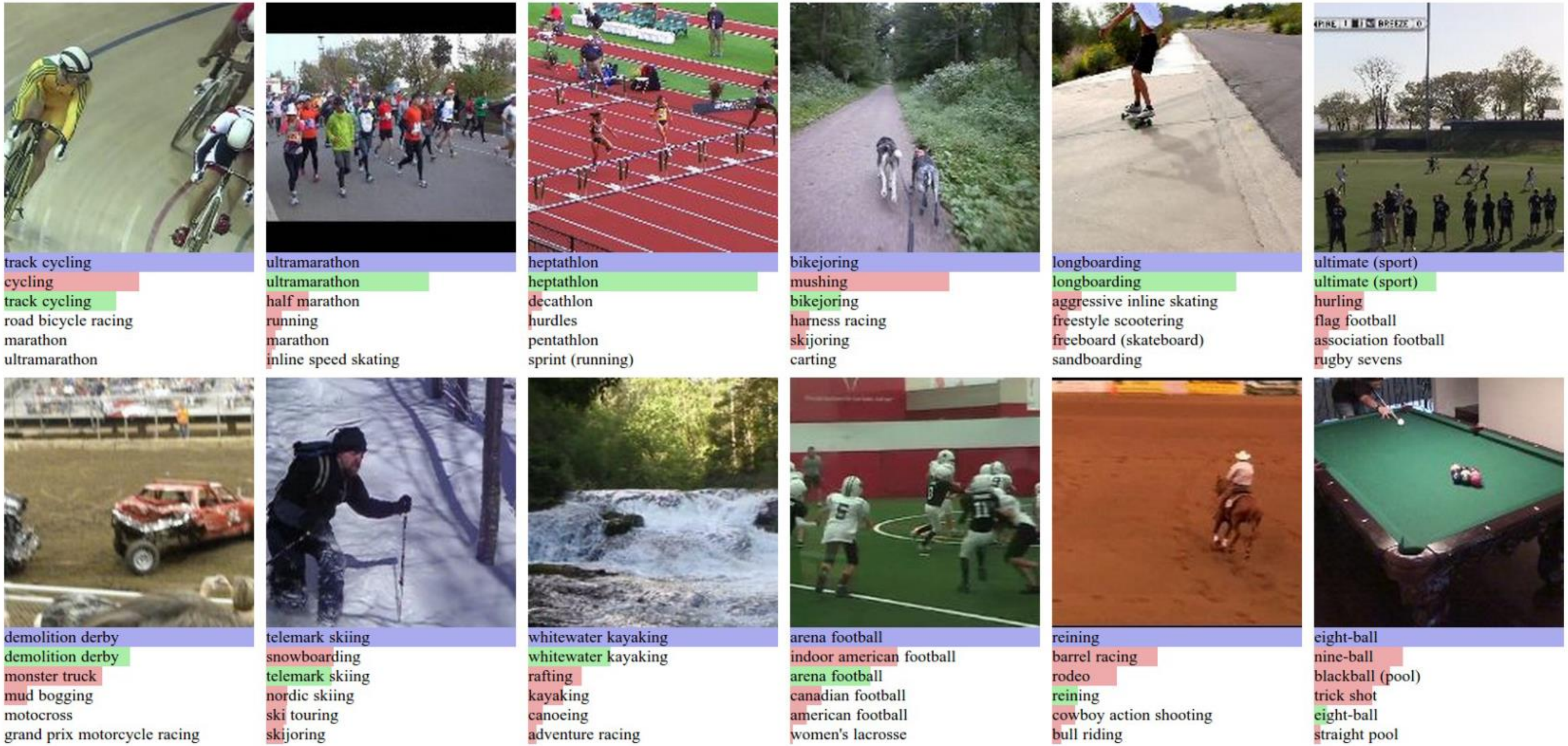


Large-scale Video Classification with Convolutional Neural Networks, Karpathy et al., 2014

Video Classification Example

GT

pred



Large-scale Video Classification with Convolutional Neural Networks, Karpathy et al., 2014

Video Classification Example

Single frame is very good and even better with a multi-resolution approach

Slow fusion works better than early and late fusion

Model	Clip Hit@1
Feature Histograms + Neural Net	-
Single-Frame	41.1
Single-Frame + Multires	**42.4**
Single-Frame Fovea Only	30.0
Single-Frame Context Only	38.1
Early Fusion	38.9
Late Fusion	40.7
Slow Fusion	**41.9**

Large-scale Video Classification with Convolutional Neural Networks, Karpathy et al., 2014

Motion

Johansson, "Visual perception of biological motion and a model for its analysis." 1973

[source]

Johansson, “Visual perception of biological motion and a model for its analysis.” 1973

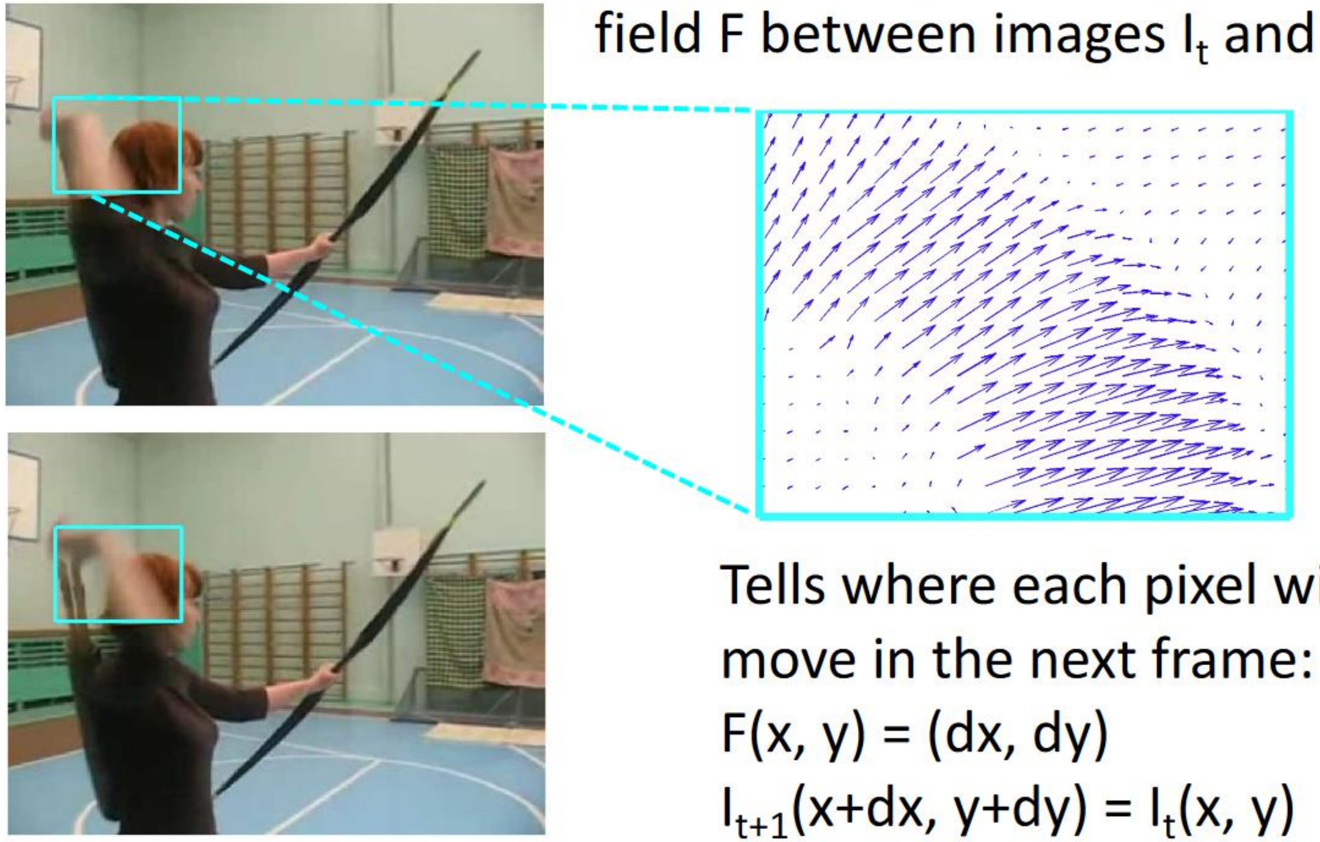
source

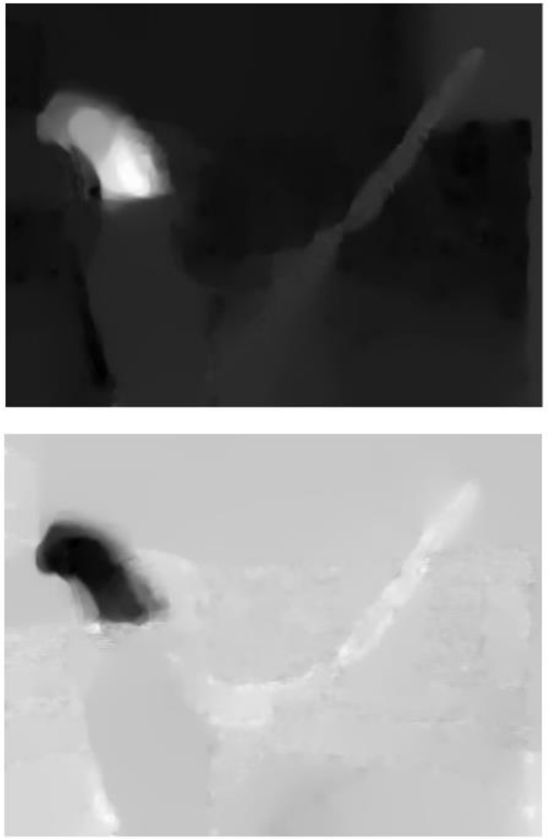
Recap: Optical Flow

Image at frame t

Optical flow gives a displacement field F between images I_t and I_{t+1}

Horizontal flow dx





Tells where each pixel will move in the next frame:

$F(x, y) = (dx, dy)$

$I_{t+1}(x+dx, y+dy) = I_t(x, y)$

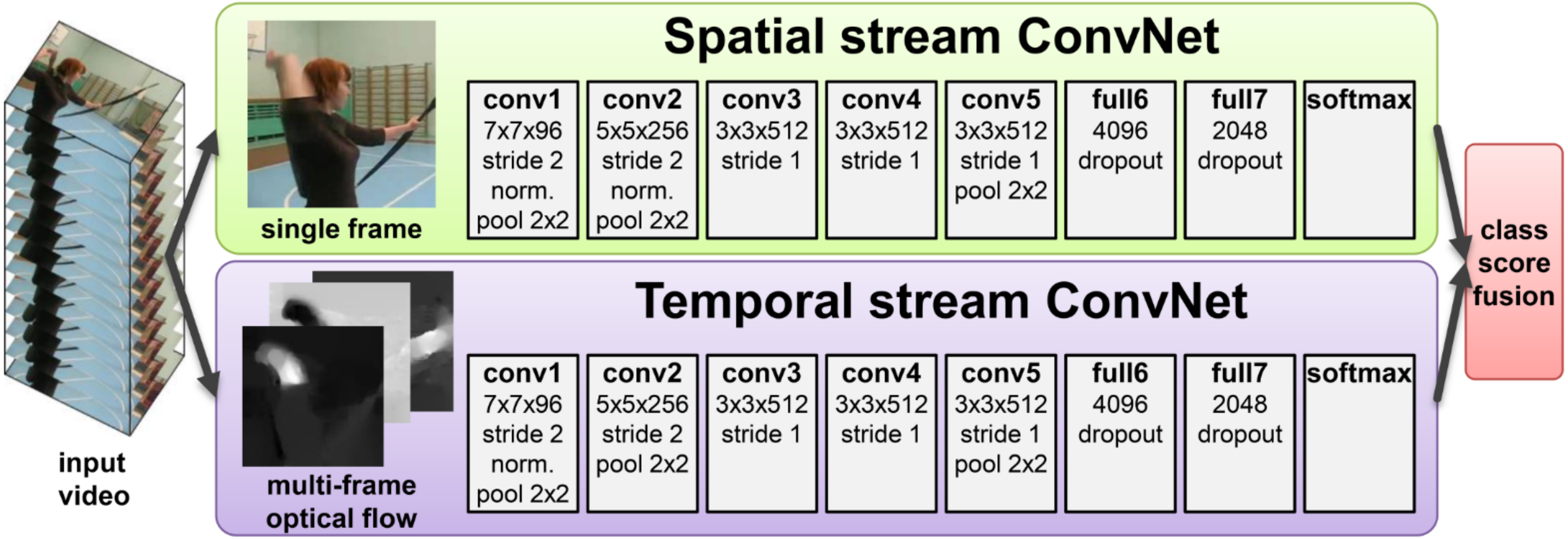
Image at frame t+1

Vertical Flow dy

Slide from J Johnson

Action Recognition with Optical Flow

Input: Single Image
3 x H x W



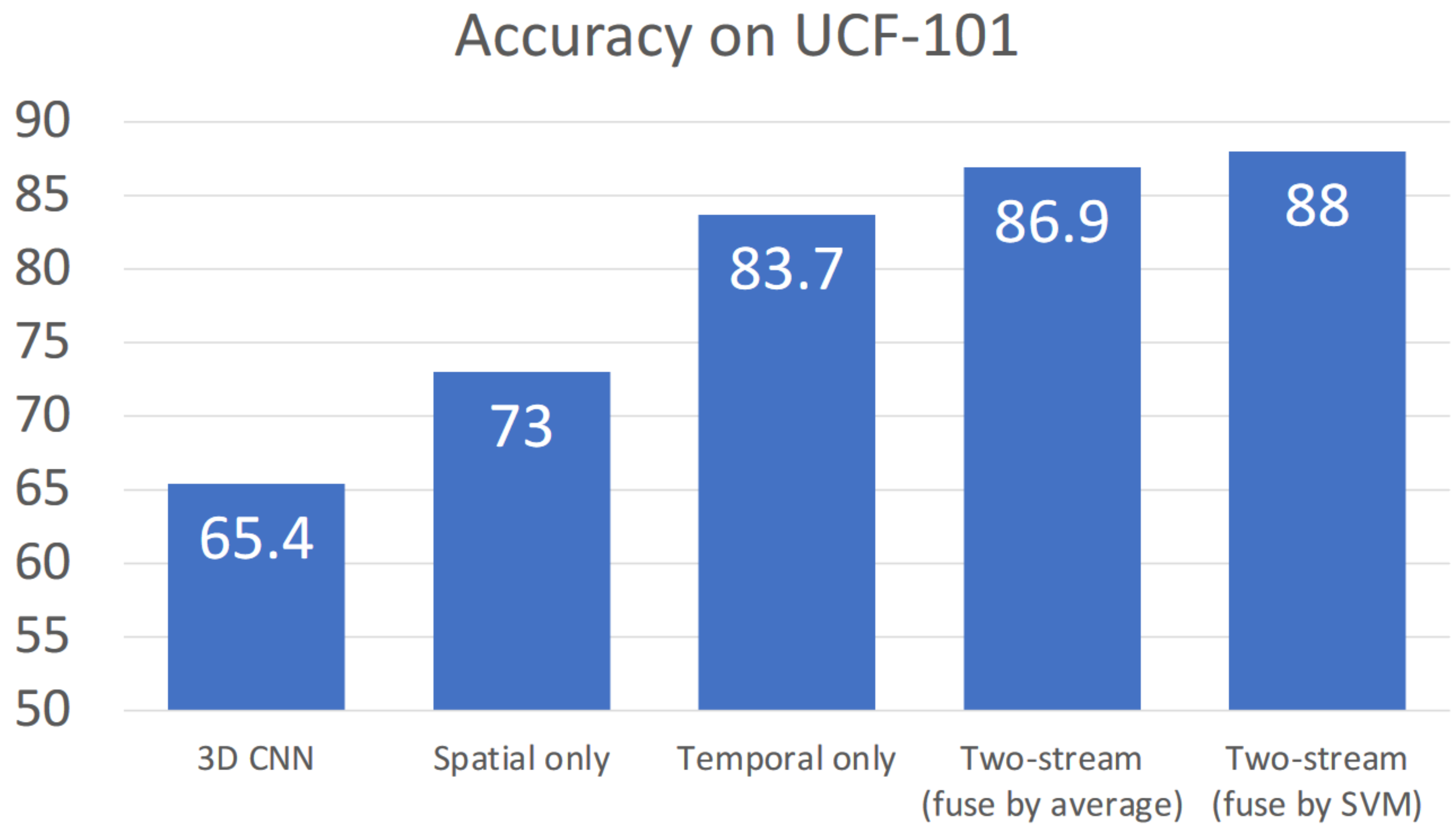
Input: Stack of optical flow:
[2*(T-1)] x H x W

Early fusion: First 2D conv processes all flow images

Simonyan and Zisserman, "Two-stream convolutional networks for action recognition in videos", NeurIPS 2014

Slide from J Johnson

Two Stream Networks



Simonyan and Zisserman, "Two-stream convolutional networks for action recognition in videos", NeurIPS 2014

Slide from J Johnson

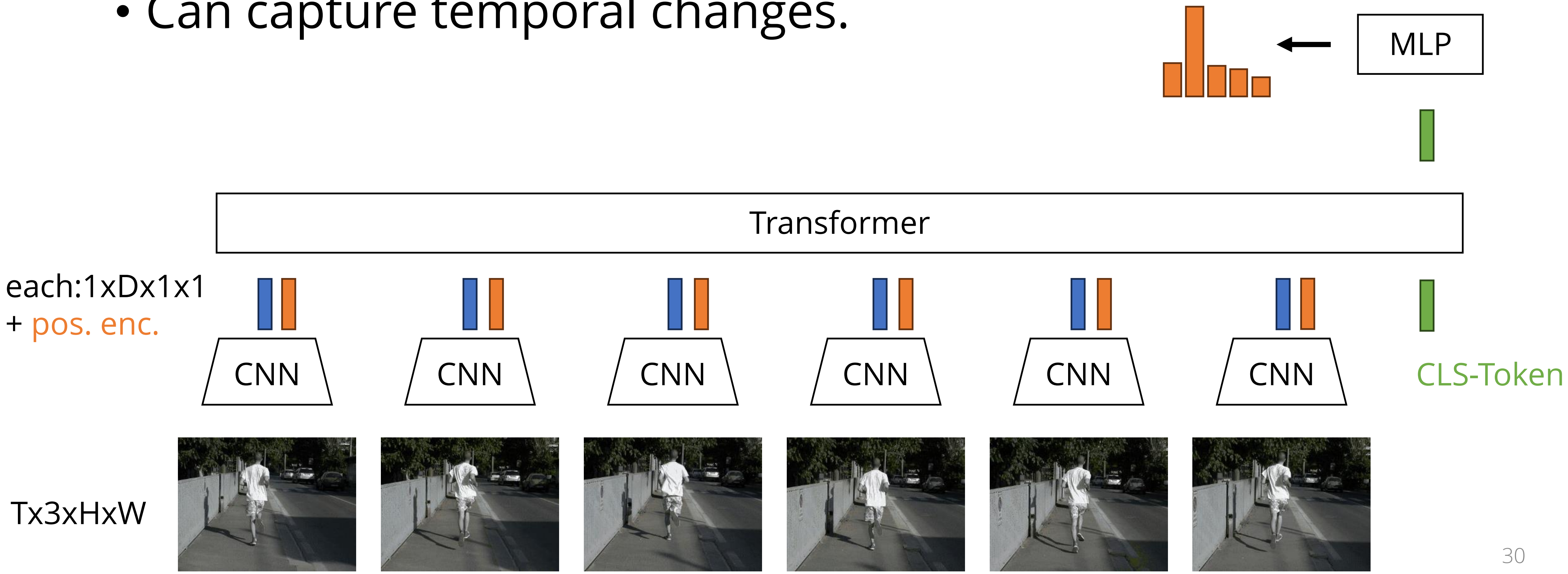
Two Stream Networks

- Can be used to fuse different modalities:
  - RGB and optical flow
  - Image and Audio
  - Image and Text
  - …

- Typically late(-ish) fusion: process each modality separately before fusing.

Longer Temporal Context

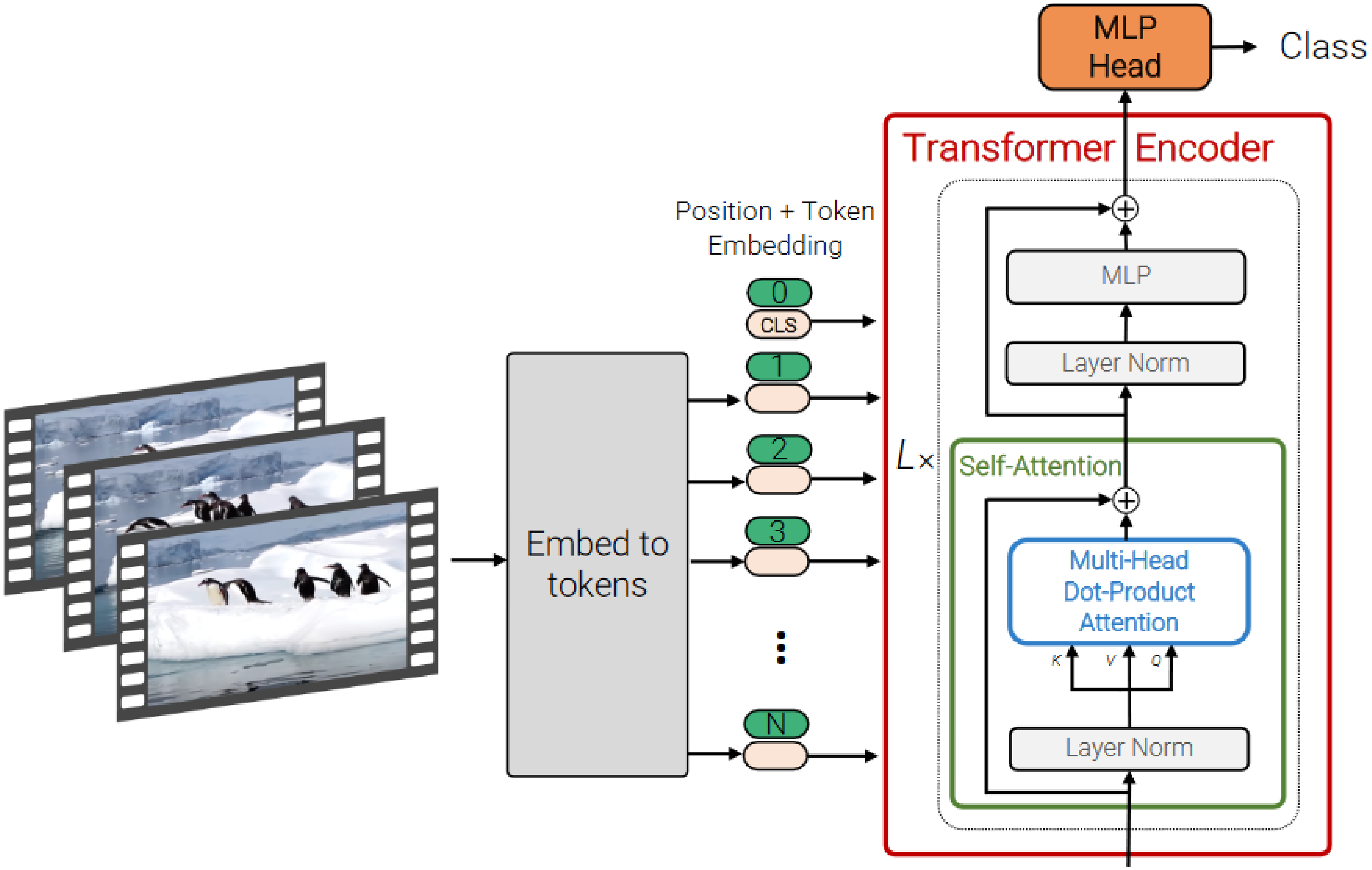
- Use a sequence model (e.g. Transformer) across time.
- Can capture temporal changes.



Longer Temporal Context

- Encoder can be per frame, or other architectures e.g. 3D CNN: produce temporal features.
- Sequence model reasons across time.
- Hybrid architecture: CNN for processing clips, transformer for processing video (composed of clips).
- Pure transformer architectures?

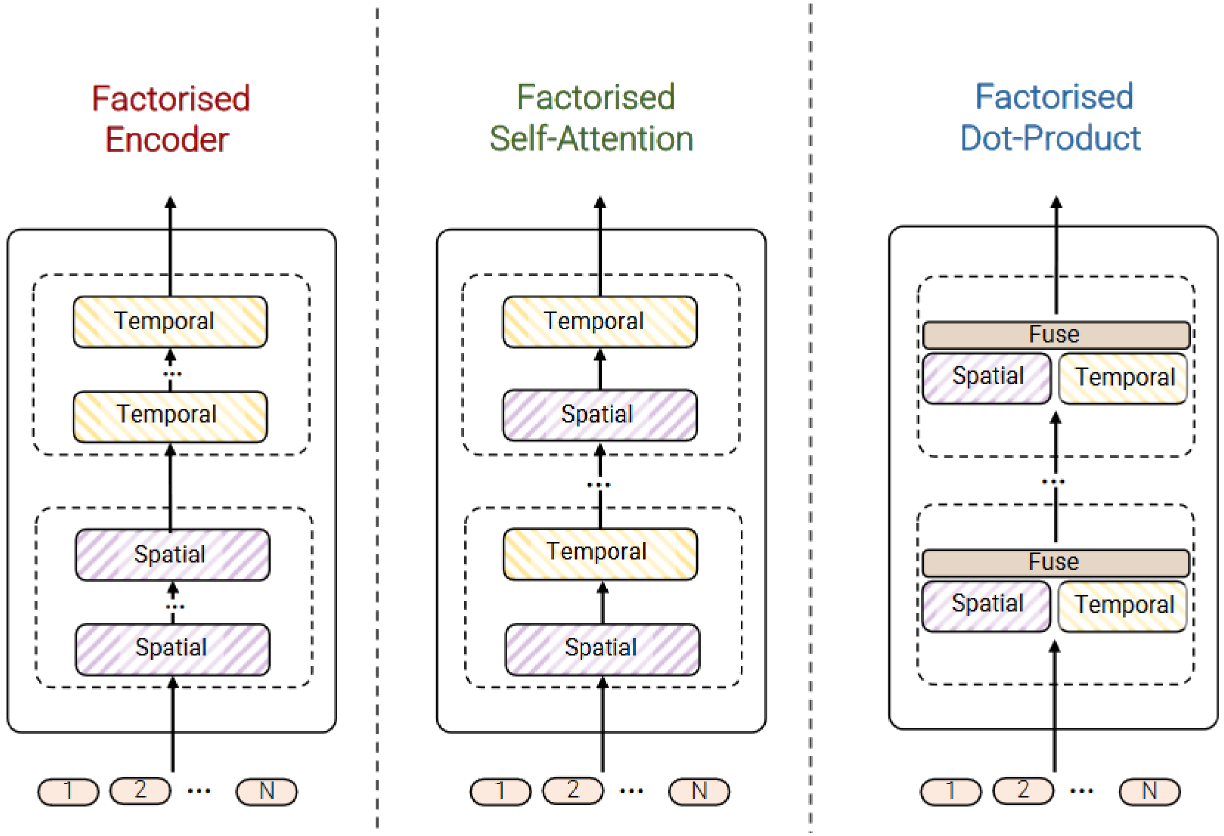
Transformer for Video Understanding



Arnab et al, “ViViT: A Video Vision Transformer”, ICCV 2021

Video Vision Transformers

- Spatio-temporal tokens: each token comes from a “patch” in space and time.
- Attention is expensive $O(n^2)$.
- Factorised attention: alternate spatial and temporal attention.



Factorised Attention

- Attention: input/output B x N x D (batch, tokens, channels)
- Samples in batch dimension are processed separately.

- Our input: B x T x D x H x W

- Idea: use batch dimension to process dimensions we do not want to include in the attention.

Temporal Attention

- Process each sample in the batch separately
- Process each spatial location separately

- Input B x T x D x H x W
    - Permute: B x H x W x T x D
    - Flatten: BHW x T x D
    - Attention: BHW x T x D -> BHW x T x D
    - Unflatten: B x H x W x T x D
    - Permute: B x T x D x H x W

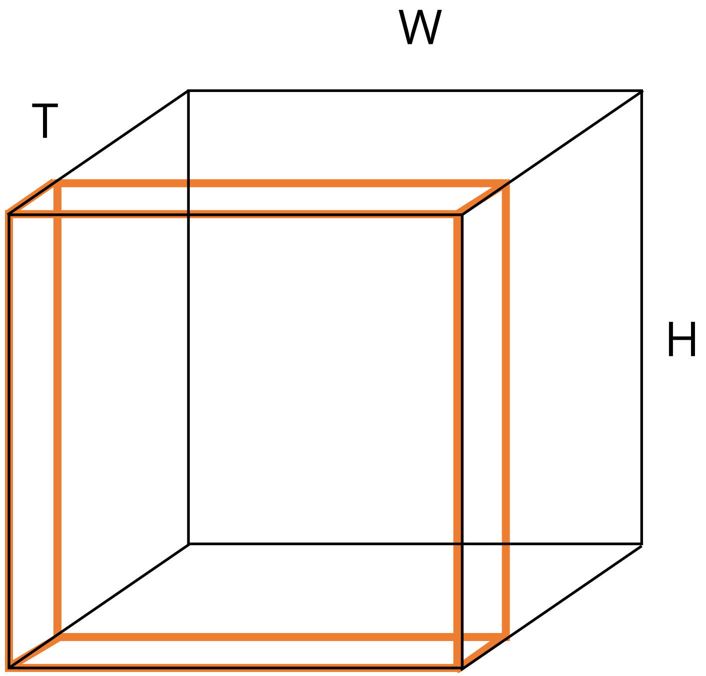
Spatial Attention

- Process each sample in the batch separately
- Process each time step separately

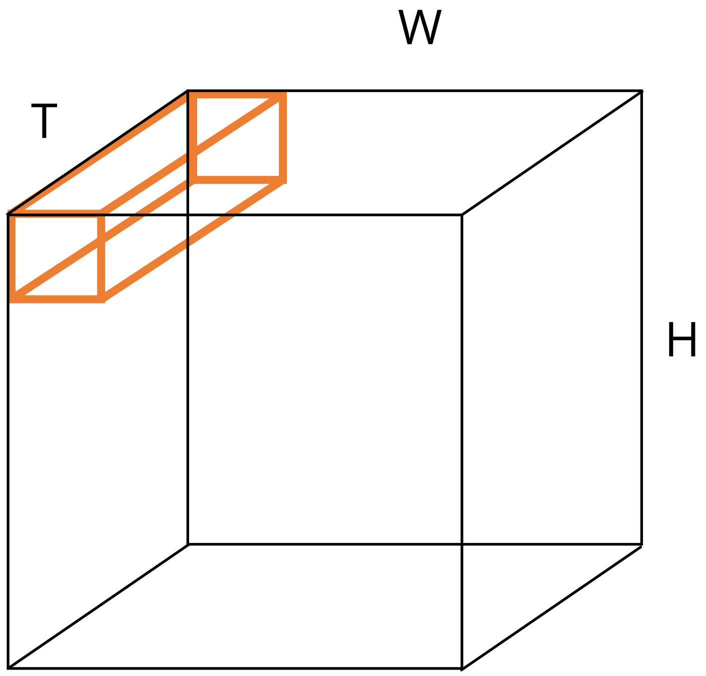
- Input B x T x D x H x W

  - Permute: B x T x H x W x D
  - Flatten: BT x HW x D
  - Attention: BT x HW x D -> BT x HW x D
  - Unflatten: B x T x H x W x D
  - Permute: B x T x D x H x W

Factorised Attention



Spatial Attention



Temporal Attention

Factorised Attention

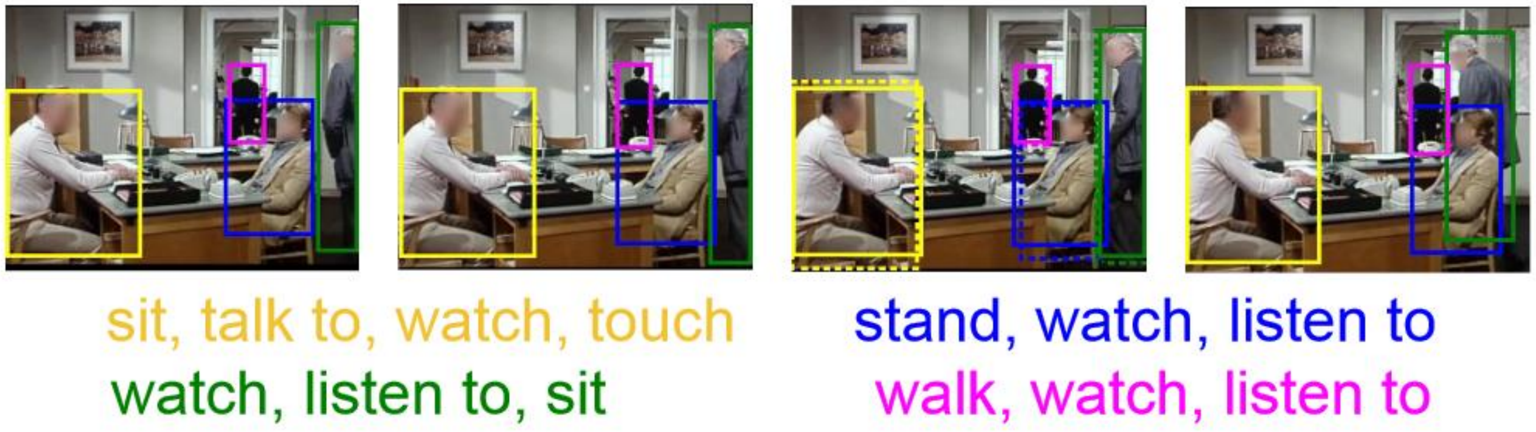
- Alternating spatial and temporal attention layers.

- Allows propagating information across time and space.

- Complexity:
  - Full attention: $\mathcal{O}\left((THW)^2\right)$
  - Factorised attention: $\mathcal{O}(T^2HW + T(HW)^2)$

Task: Spatio-Temporal Detection

- Detect objects in a video and predict their actions.
- Tubelets: bounding box with time.



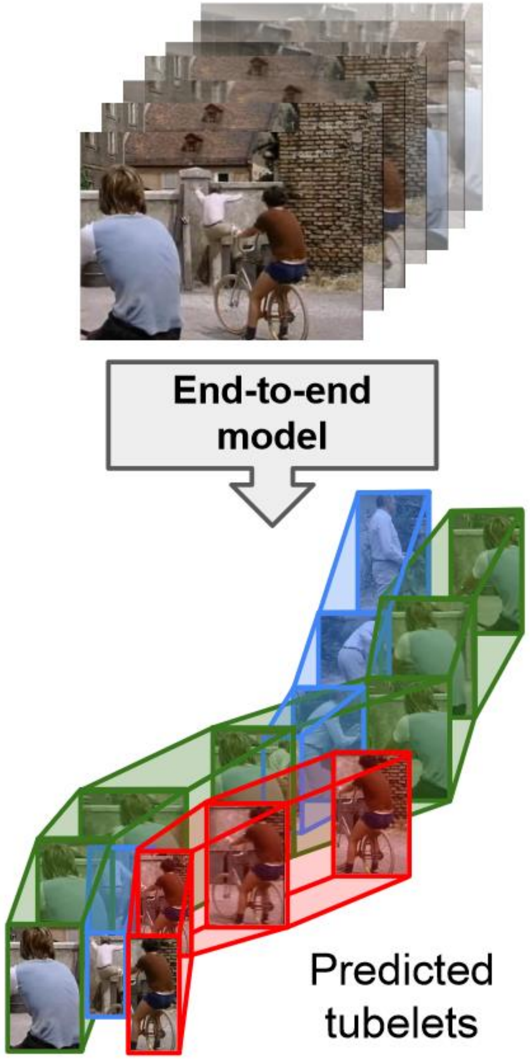
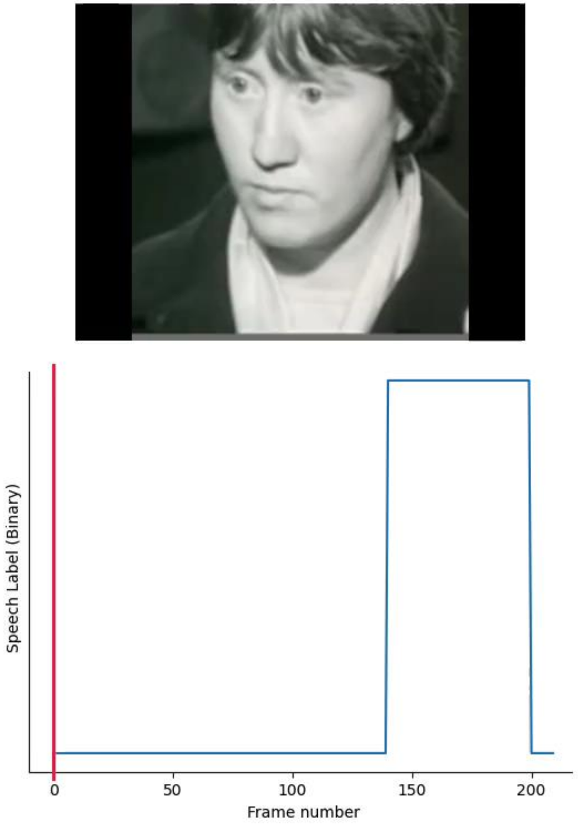


Image source

Task: Lip Reading

Step 1: Visual Speech Detection



Sub-word Level Lip Reading With Visual Attention, Prajwal et al., 2022

Lip Reading with Attention

Figure 2. Visualization of the visual attention masks a from the VTP module superimposed on the input frames that produce them. The video clips used here are random samples from the LRS3 dataset. It is evident that the model follows the more discriminative mouth region.

Sub-word Level Lip Reading With Visual Attention, Prajwal et al., 2022

Task: Audio Description

- Multi-modal task: audio-visual input -> text
- Difficult: long-range context understanding

AutoAD: Movie Description in Context, Han et al., 2021

Multi-Modal Learning

- Often, we have multiple modalities: image, audio, text, sensor signal, 3D, temperature, etc.
- Fusion-based architectures are good to combine different input types.
- Process each modality separately into a common shape.
- Fuse and process jointly.

Processing Videos

- Expensive task even after all the subsampling.
- High compute and memory cost.
- Many tasks: image models work surprisingly well.
- Architectures: combine image and time understanding.