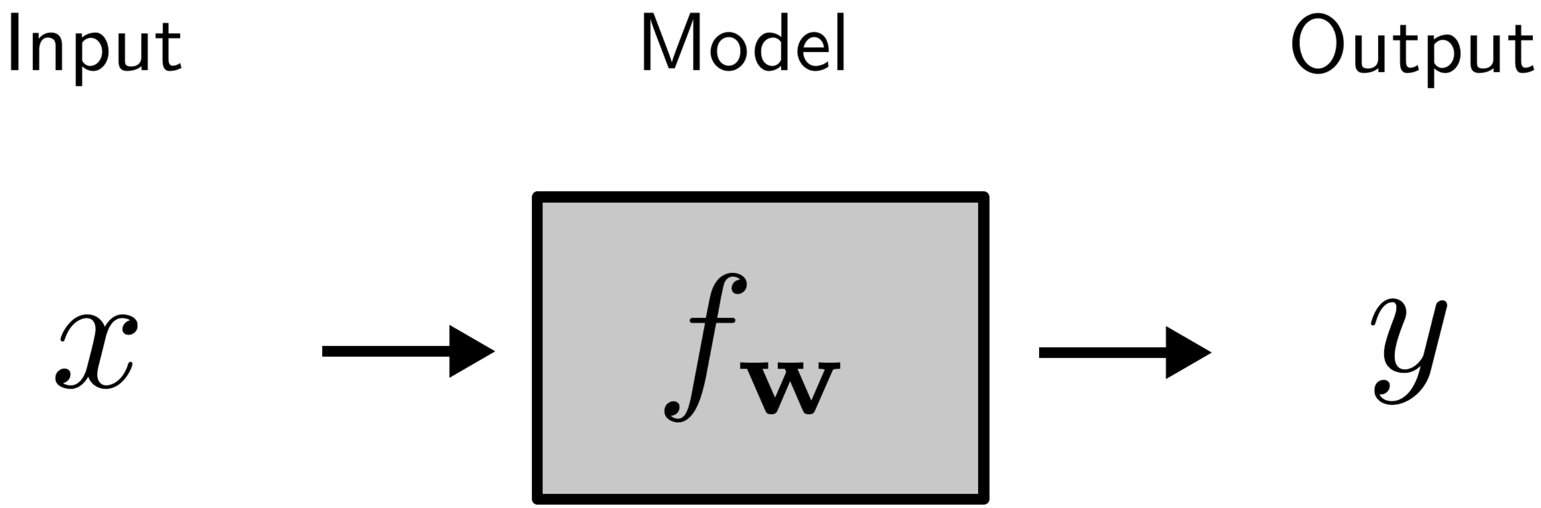
From Action Recognition to Action Anticipation

Oswald Lanz

Univ. of Bolzano

https://vision.inf.unibz.it

Supervised Learning



- **Learning:** Estimate parameters $\mathbf{w}$ from training data $\{(x_i, y_i)\}_{i=1}^{N}$
- **Inference:** Make novel predictions: $y = f_{\mathbf{w}}(x)$

Classification

Input

Model

$f_{\mathbf{w}}$

Output

"Beach"

- **Mapping:** $f_{\mathbf{w}} : \mathbb{R}^{W \times H} \rightarrow \{\text{“Beach”}, \text{“No Beach”}\}$

Key Moment in History of Deep Learning

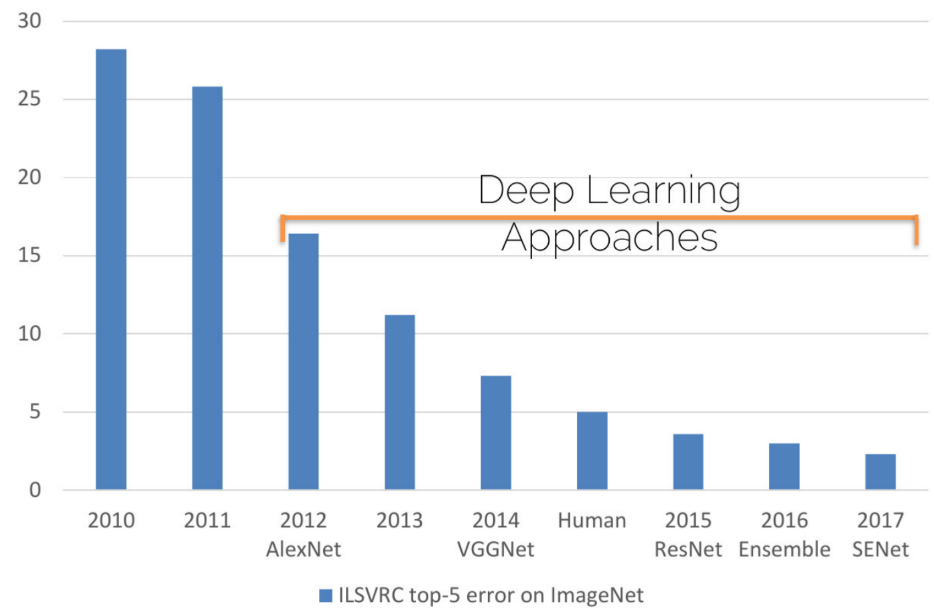
2009-2012: ImageNet and AlexNet

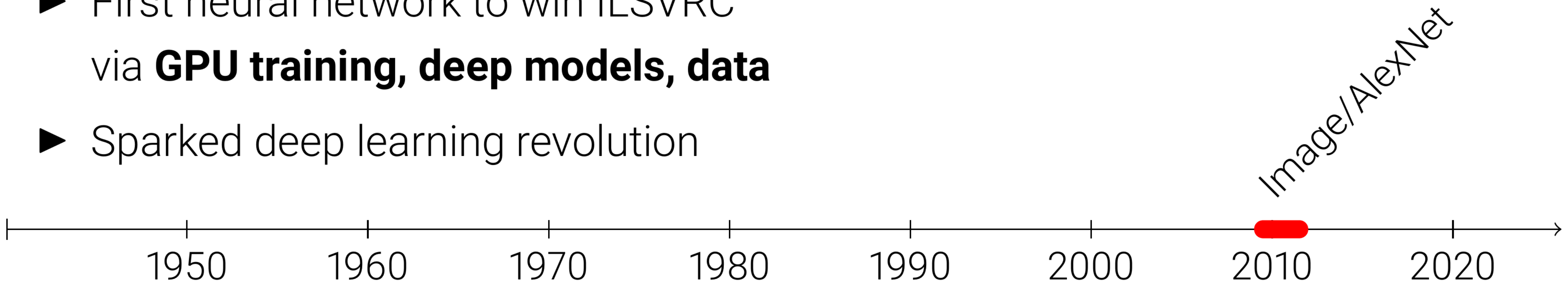
ImageNet

- Recognition benchmark (ILSVRC)
- 10 million annotated images
- 1000 categories

AlexNet

- First neural network to win ILSVRC via **GPU training, deep models, data**
- Sparked deep learning revolution





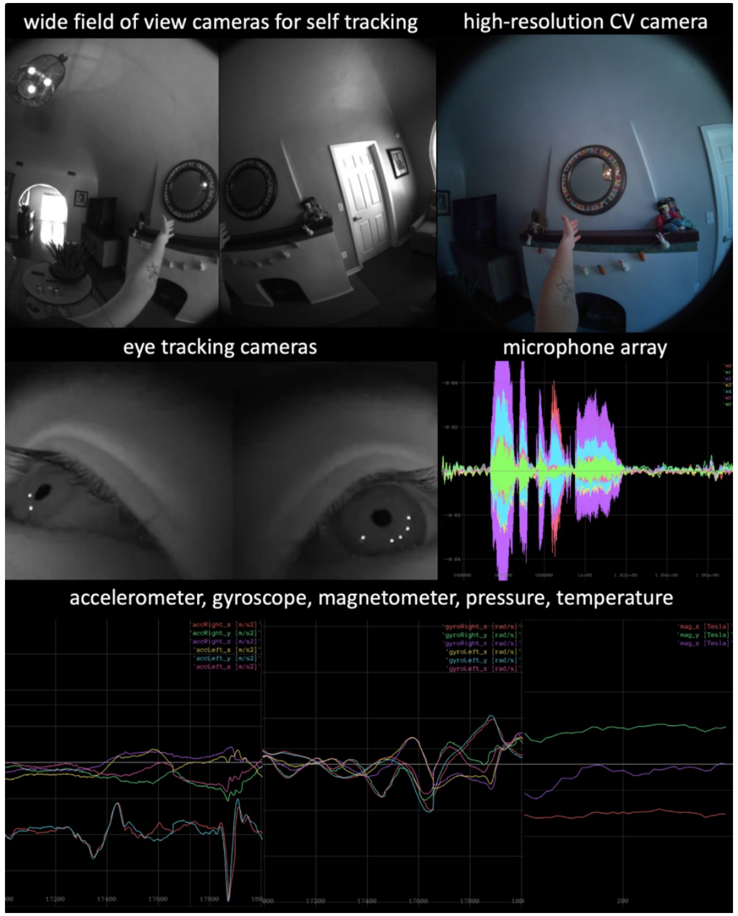
Krizhevsky, Sutskever, Hinton. ImageNet classification with deep convolutional neural networks. NIPS, 2012.

Video Action Classification

Input	Model	Output
	$f_{\mathbf{w}}$	“laugh”

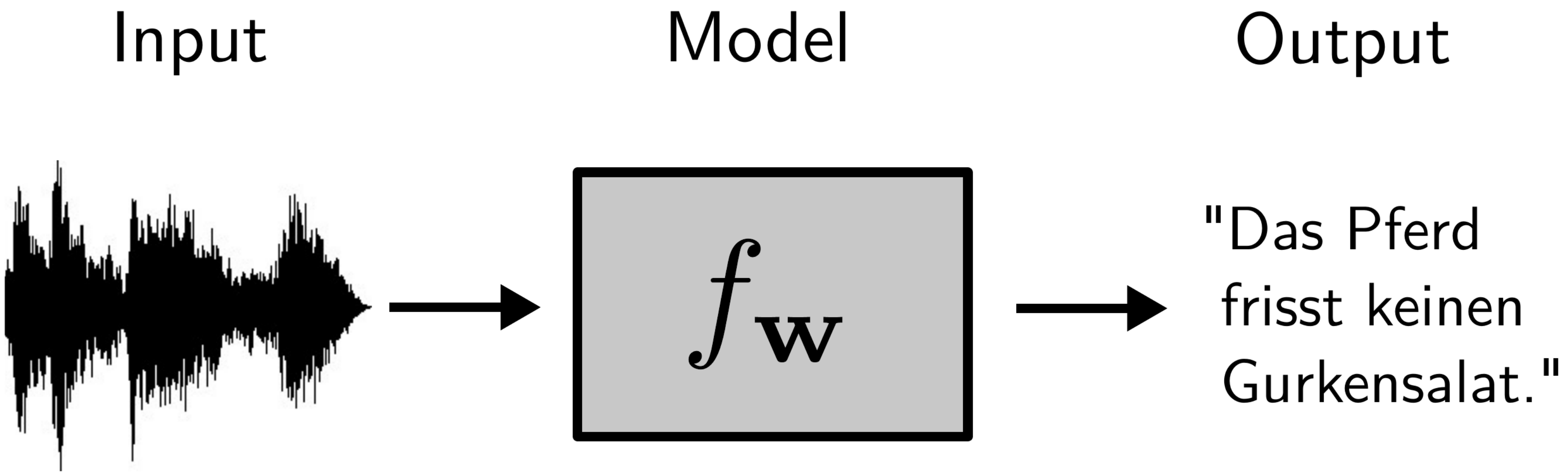
- **Mapping:** $f_{\mathbf{w}} : \mathbb{R}^{W \times H \times T} \rightarrow \{\text{"run"}, \text{"laugh"}, \text{"dive"}, \text{"eat"}, \ldots\}$

Multimodal Action Classification

Input	Model	Output
	$f_{\mathbf{w}}$	“pointing”

Meta Project Aria

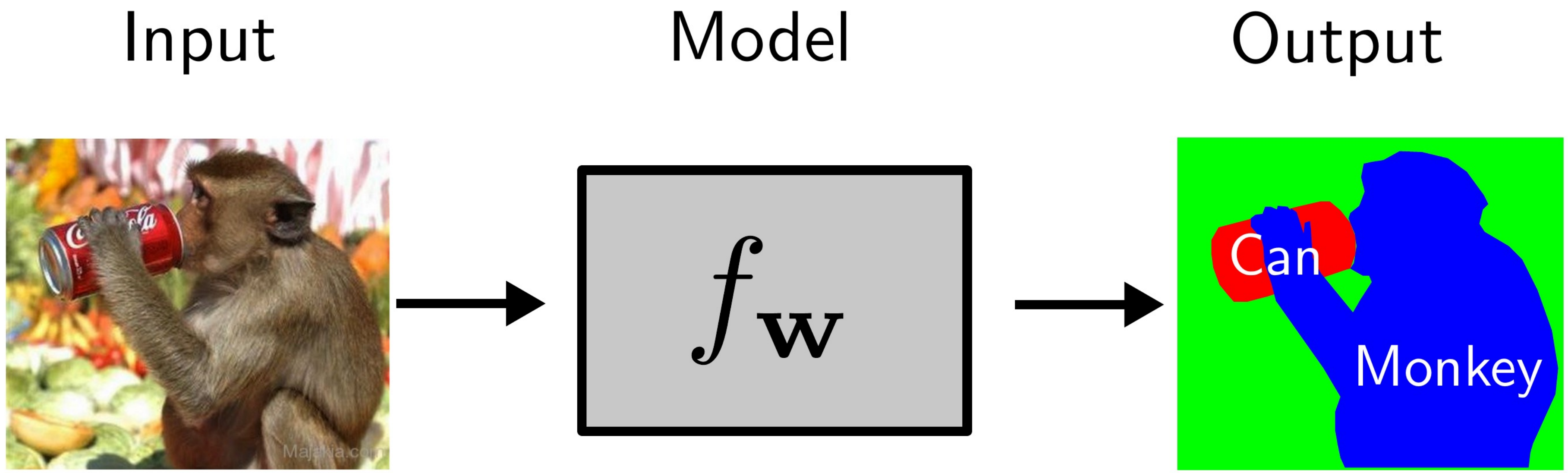
Structured Prediction



- **Mapping:** $f_{\mathbf{w}} : \mathbb{R}^N \rightarrow \{1, \dots, C\}^M$

Actions on Objects

Structured Prediction

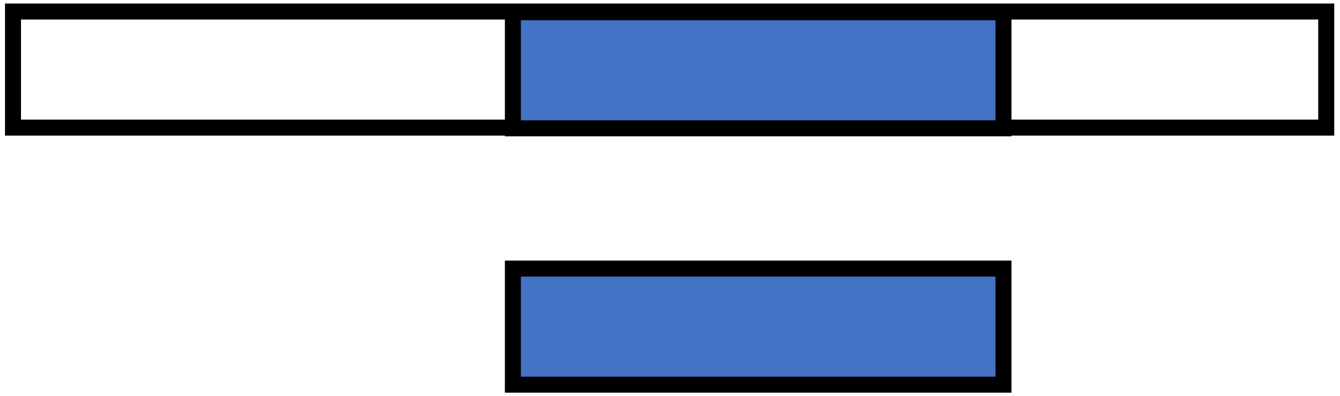


- **Mapping:** $f_{\mathbf{w}} : \mathbb{R}^{W \times H} \rightarrow \{1, \ldots, C\}^{W \times H}$

Action Detection/Localization

Input | **Model** | **Output**

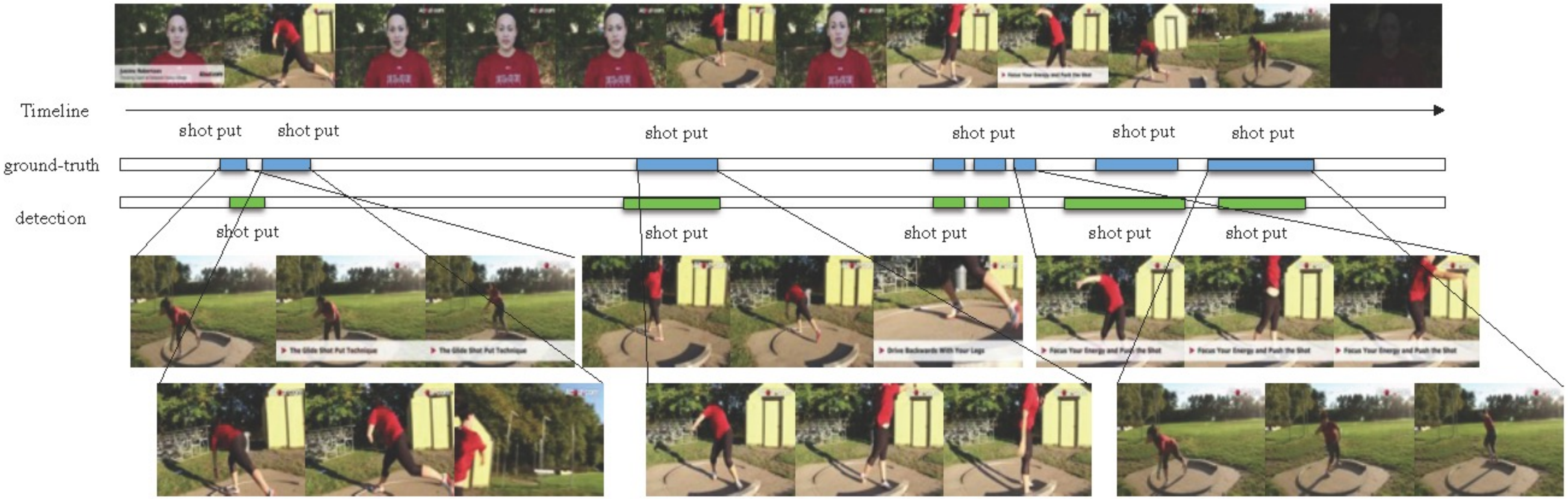
untrimmed video → $f_{\mathbf{w}}$ → action label + start-end time (+ bounding boxes/tracks)



Untrimmed Video Classification

Trimmed Video Classification

Action Detection/Localization



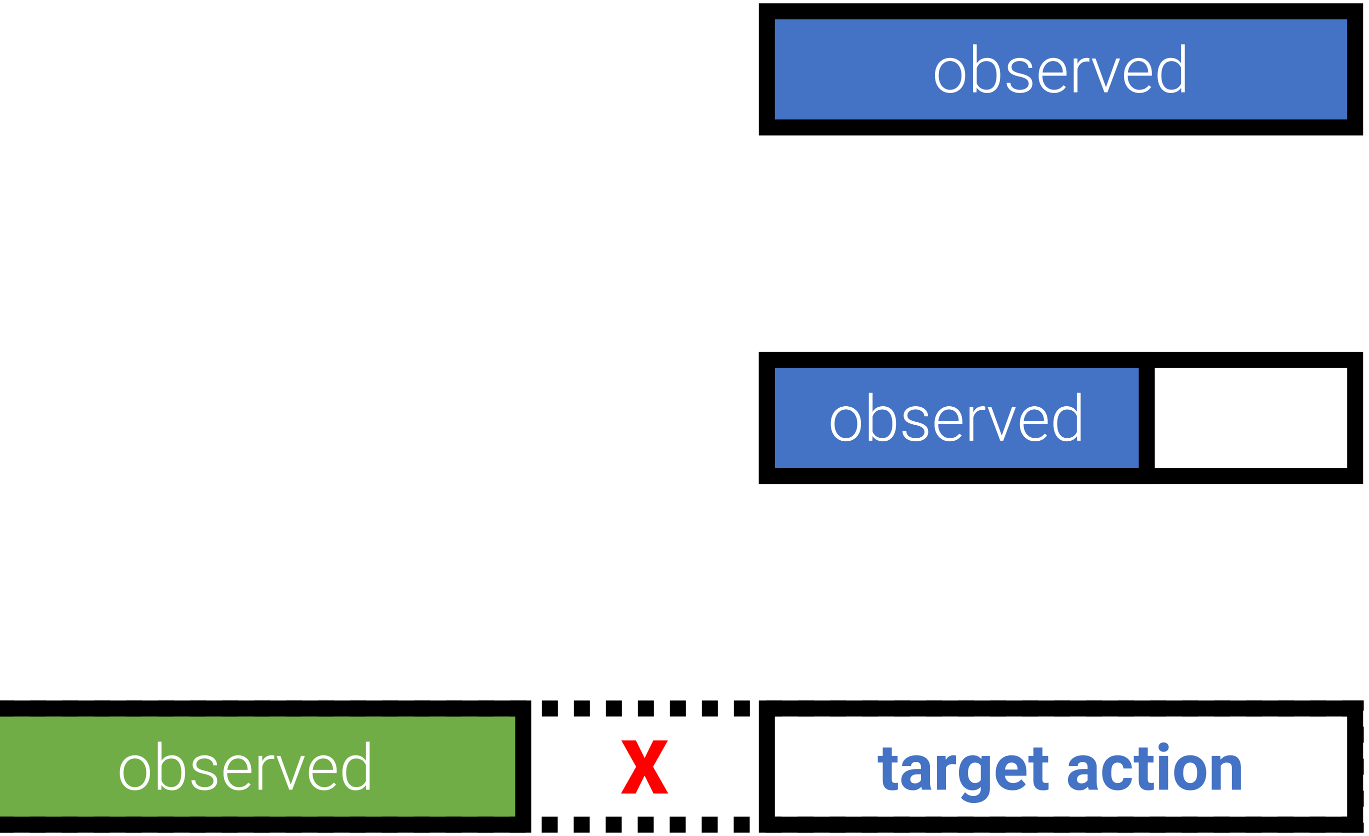
Action Segmentation

Input	Model	Output
untrimmed video	$f_{\mathbf{w}}$	action label + confidence at every frame

Untrimmed Video Classification

Untrimmed Video Segmentation

Early Action Recognition - Action Anticipation/Prediction



Action Recognition (= Trimmed Video Classification with Action Labels)

Early Action Recognition

Action Anticipation/Prediction

Action Recognition Models

Video Action Classification

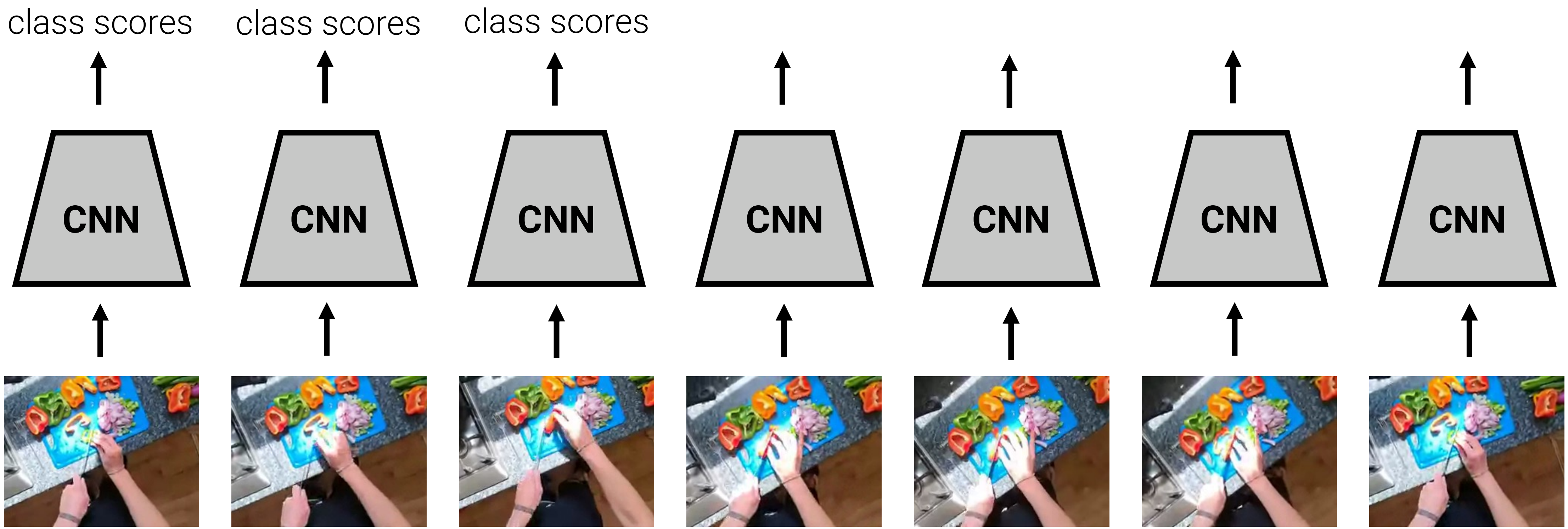
Input	Model	Output
	$f_{\mathbf{w}}$	"laugh"

Video as a **sequence/set of frames** or as a **space-time volume** ?

Single Frame CNN

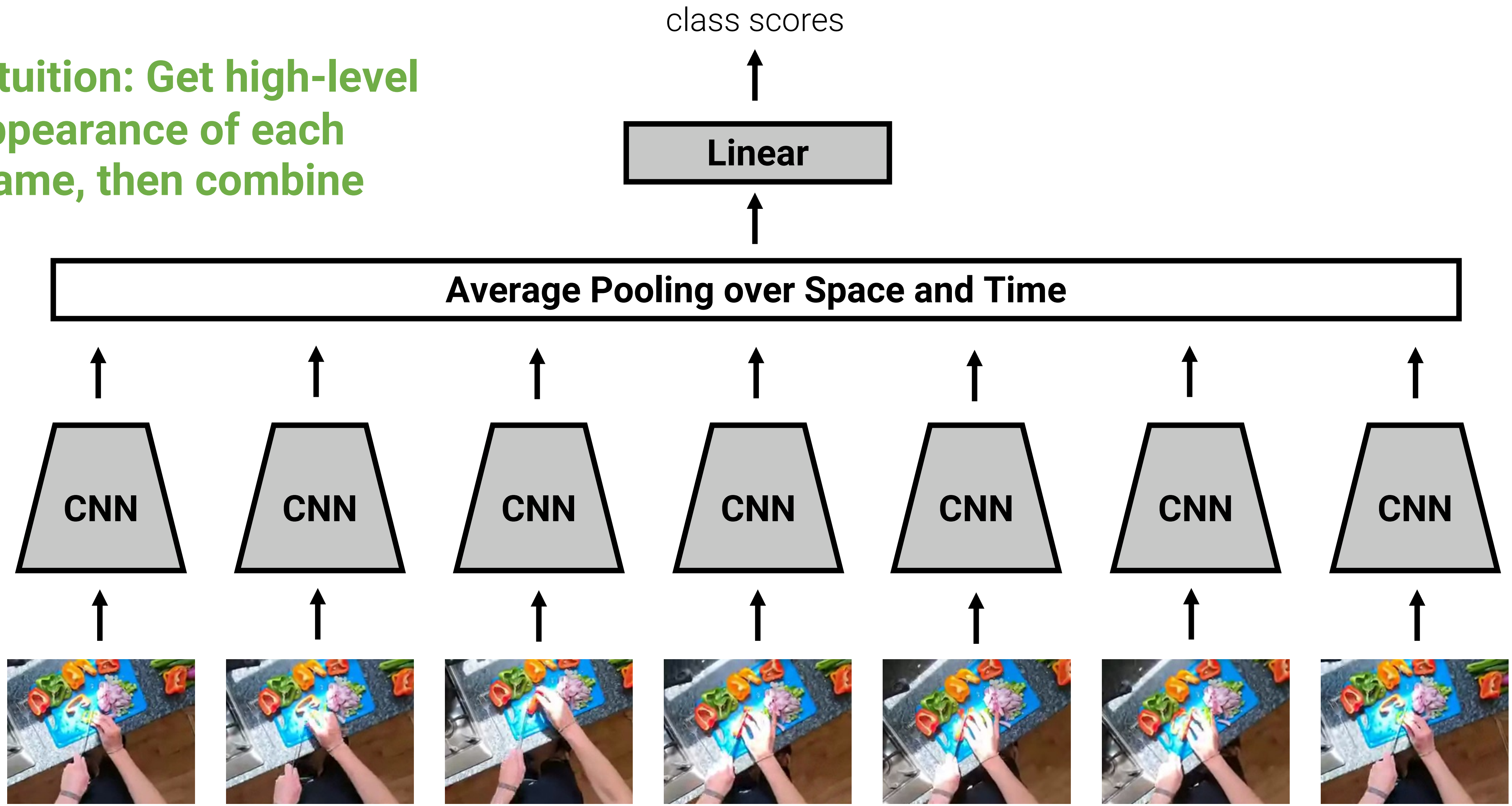
Simple idea: Train normal CNN to classify frames independently
(Average predicted probs at test-time)

Often a very **strong baseline** for video classification



Late Fusion

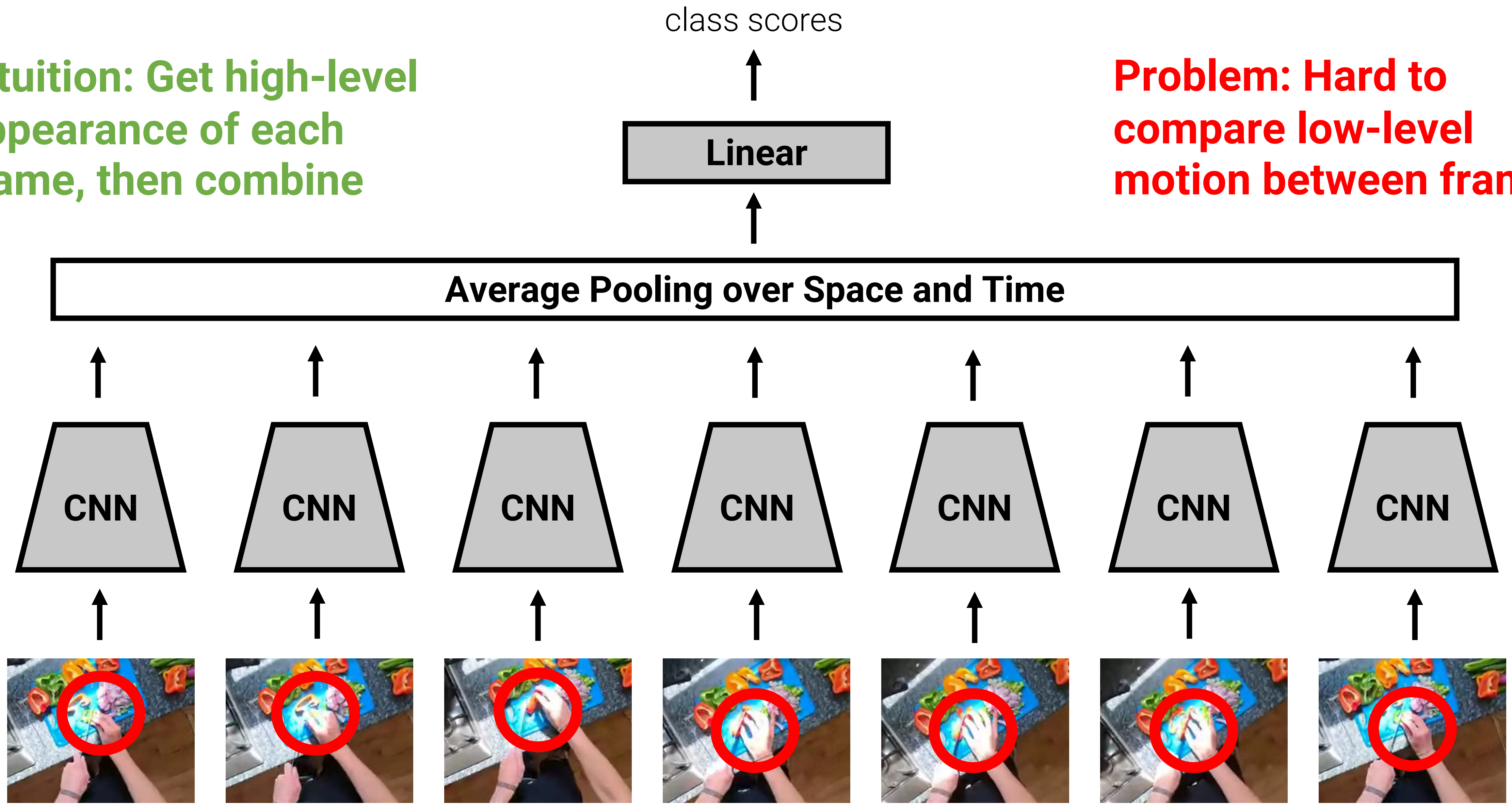
Intuition: Get high-level appearance of each frame, then combine



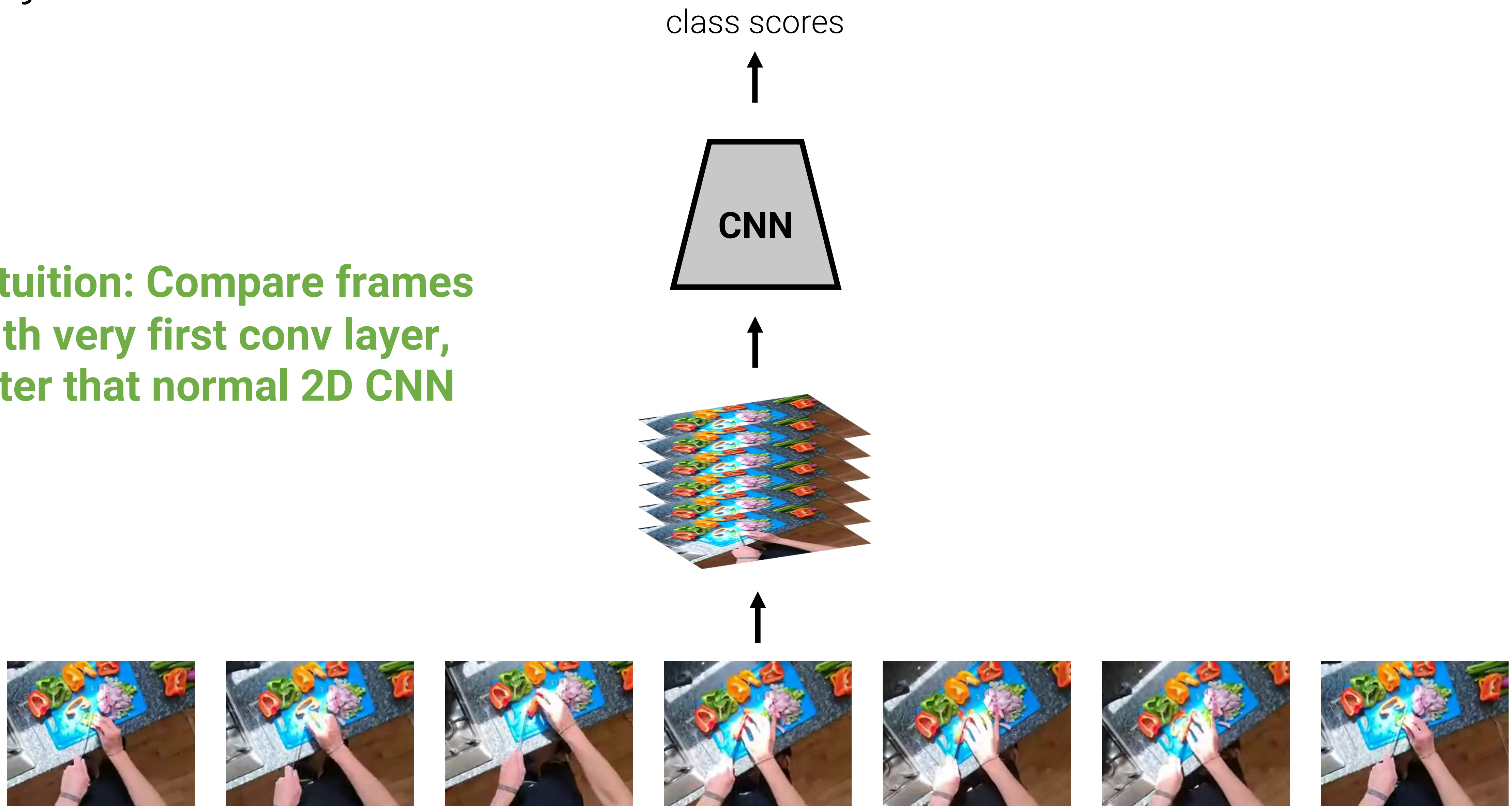
Late Fusion

Intuition: Get high-level appearance of each frame, then combine

Problem: Hard to compare low-level motion between frames

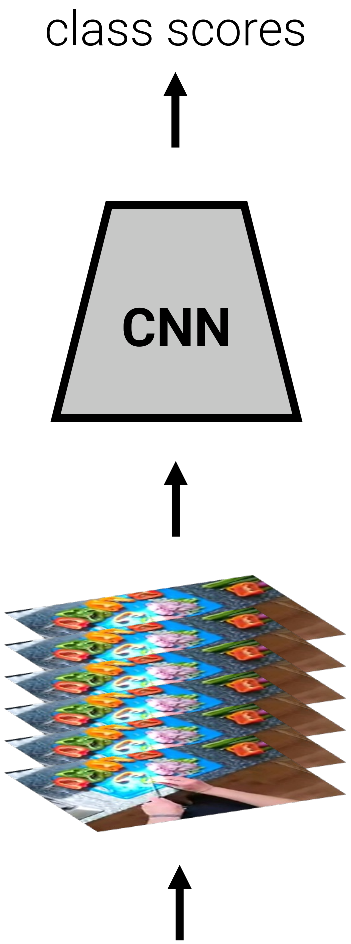


Early Fusion



Intuition: Compare frames with very first conv layer, after that normal 2D CNN

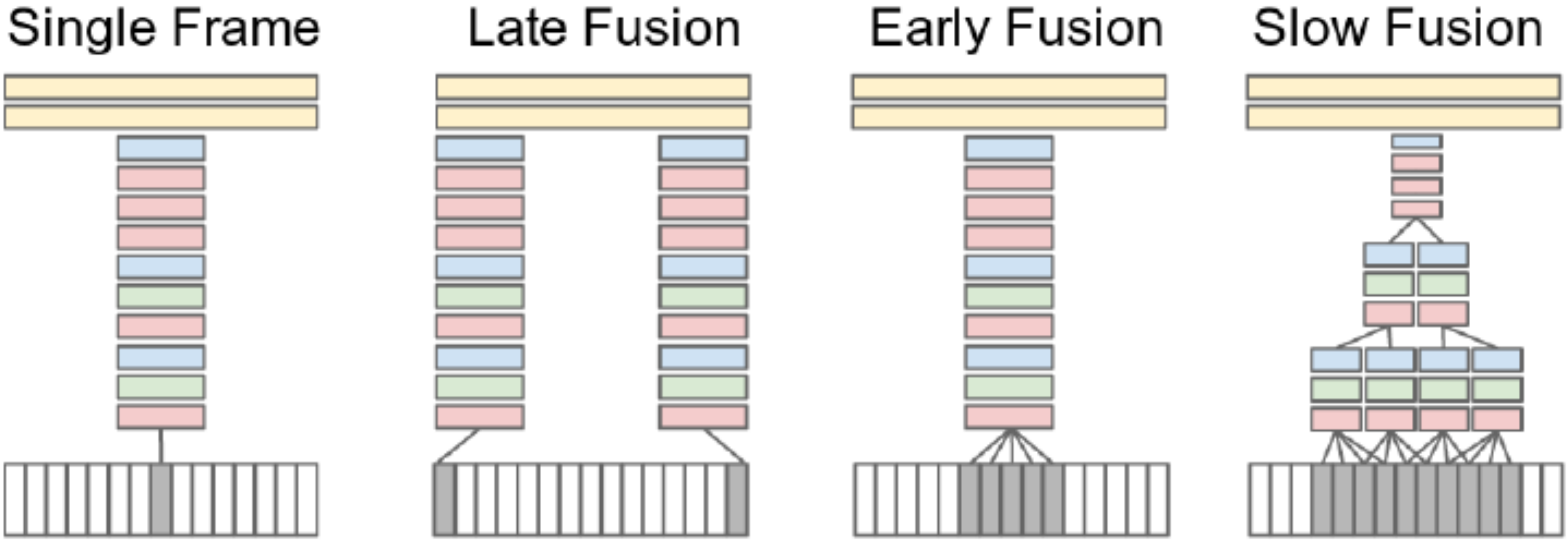
Early Fusion



Intuition: Compare frames with very first conv layer, after that normal 2D CNN

Problem: One layer of temporal processing may not be enough

Slow Fusion



Karpathy et.al, “Large-Scale Video Classification with Convolutional Neural Networks”. CVPR 2014
.

Example Video Dataset: Sports-1M

- 1 million YouTube videos annotated with labels for 487 different types of sports

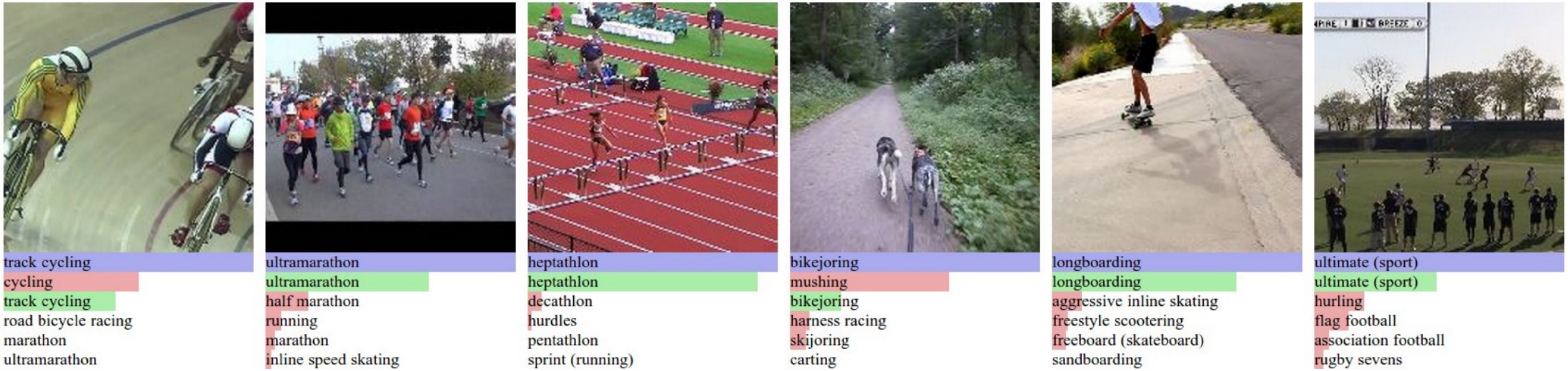


Figure 4: Predictions on Sports-1M test data. Blue (first row) indicates ground truth label and the bars below show model predictions sorted in decreasing confidence. Green and red distinguish correct and incorrect predictions, respectively.

Karpathy et.al, "Large-Scale Video Classification with Convolutional Neural Networks". CVPR 2014

Video Classification with 2D CNN

- 1 million YouTube videos annotated with labels for 487 different types of sports

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

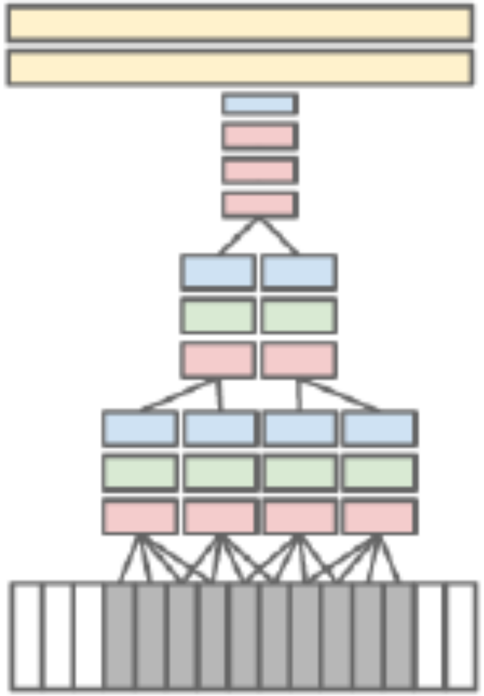
Table 1: Results on the 200,000 videos of the Sports-1M test set. Hit@k values indicate the fraction of test samples that contained at least one of the ground truth labels in the top k predictions.

Karpathy et.al, “Large-Scale Video Classification with Convolutional Neural Networks”. CVPR 2014

.

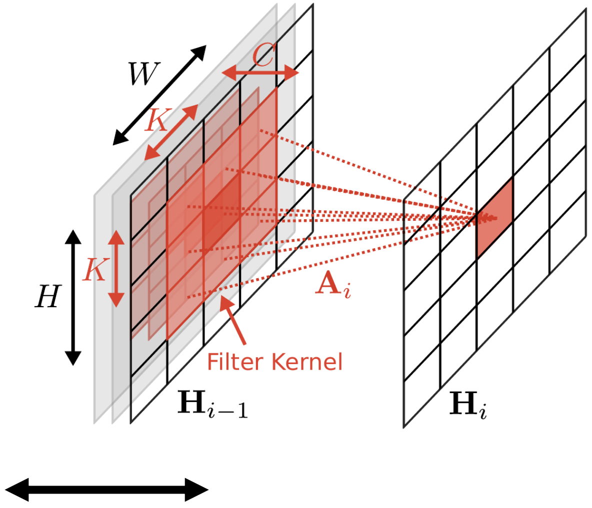
Slow Fusion

= Better performance = Good news.



We inherit **spatial shift-equivariance** from Conv2D layers.

But what about **temporal shift-equivariance** (same local motion happening sooner or later in the video) ?

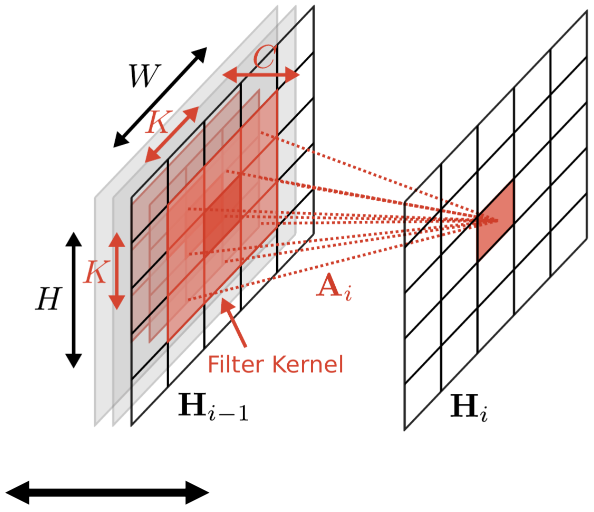


No weight sharing across time, hence no temporal shift-invariance

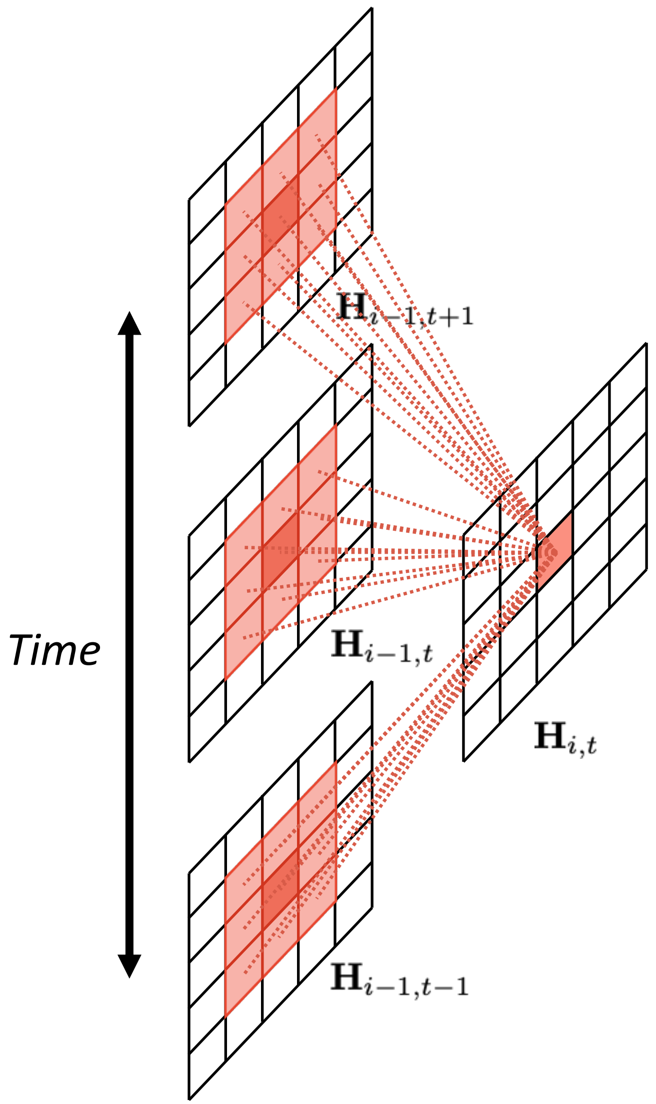
Needs to learn separate filters for same local motion at different times in the clip

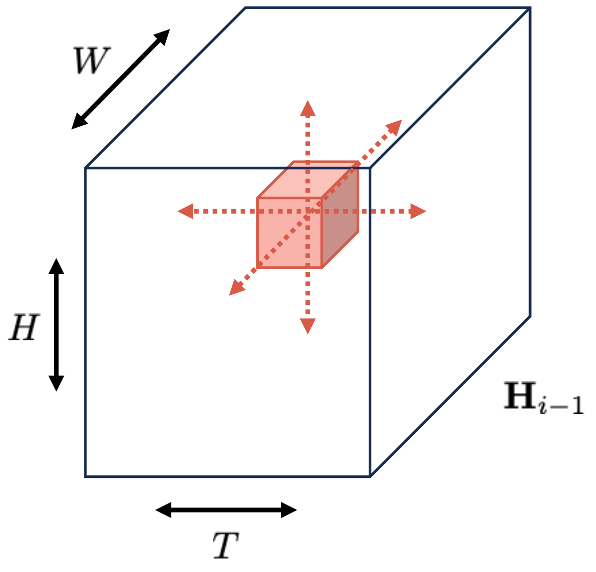
Karpathy et.al, "Large-Scale Video Classification with Convolutional Neural Networks". CVPR 2014

3D Conv (3D CNN)



channels = *Time*





Temporal shift-invariant since each filter slides over time

Early Fusion (2D Conv)

3D CNN on Space-Time (3D Conv)

C3D: The VGG of 3D CNNs

3D CNN that uses all 3x3x3 conv and 2x2x2 pooling (except Pool1 which is 1x2x2)

Released model pretrained on Sports-1M: Many people used this as a video feature extractor

Problem: 3x3x3 conv is very expensive

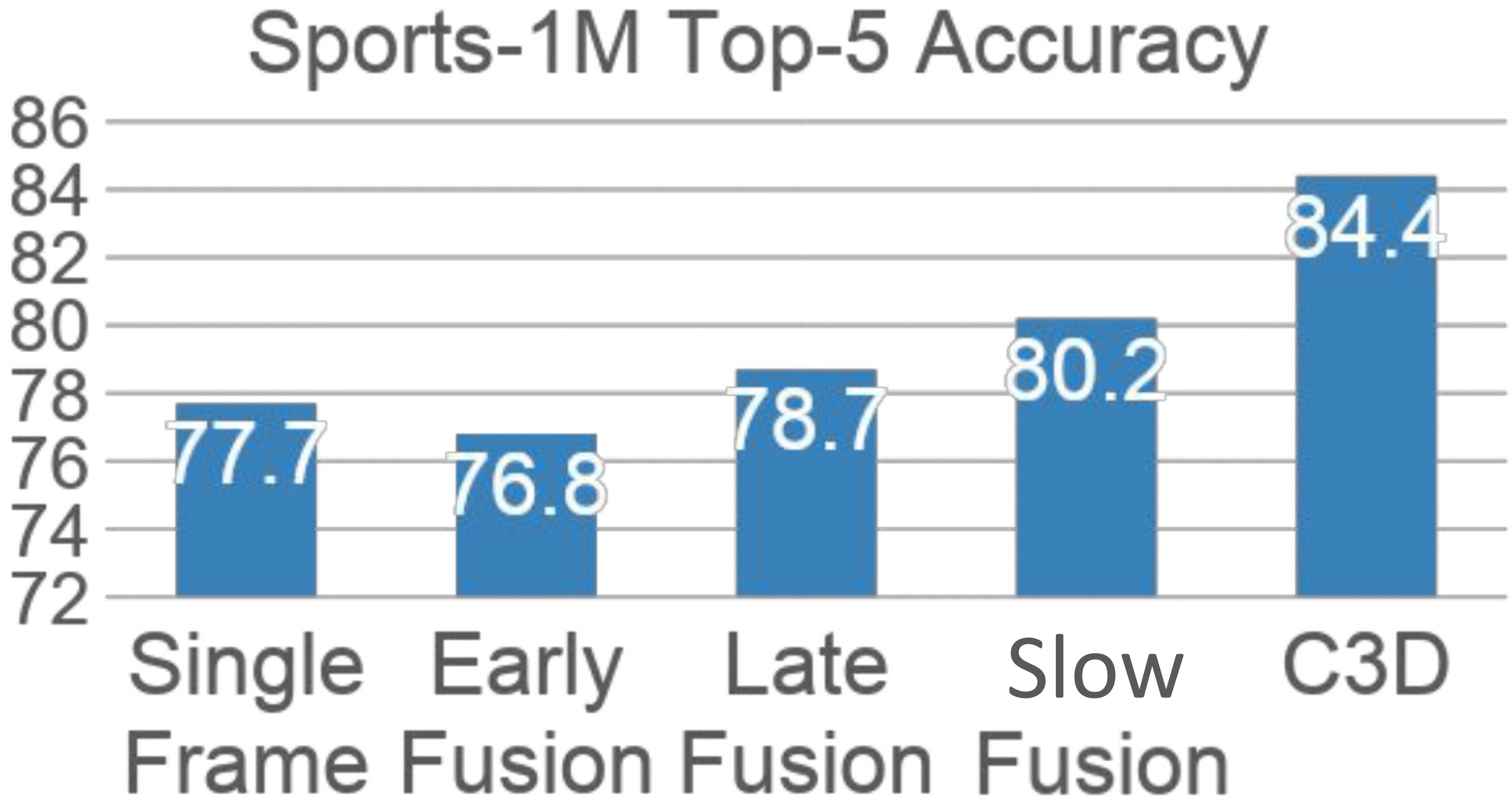
AlexNet: 0.7 GFLOP
VGG-16: 13.6 GFLOP
C3D: **39.6 GFLOP (2.9x of VGG)**

Layer	Size	MFLOPs
Input	3 x 16 x 112 x 112	
Conv1 (3x3x3)	64 x 16 x 112 x 112	1.04
Pool1 (1x2x2)	64 x 16 x 56 x 56	
Conv2 (3x3x3)	128 x 16 x 56 x 56	11.10
Pool2 (2x2x2)	128 x 8 x 28 x 28	
Conv3a (3x3x3)	256 x 8 x 28 x 28	5.55
Conv3b (3x3x3)	256 x 8 x 28 x 28	11.10
Pool3 (2x2x2)	256 x 4 x 14 x 14	
Conv4a (3x3x3)	512 x 4 x 14 x 14	2.77
Conv4b (3x3x3)	512 x 4 x 14 x 14	5.55
Pool4 (2x2x2)	512 x 2 x 7 x 7	
Conv5a (3x3x3)	512 x 2 x 7 x 7	0.69
Conv5b (3x3x3)	512 x 2 x 7 x 7	0.69
Pool5	512 x 1 x 3 x 3	
FC6	4096	0.51
FC7	4096	0.45
FC8	C	0.05

Tran et.al, “Learning Spatio-temporal Features with 3D Convolutional Networks”. ICCV 2015

Slide credit: Justin Johnson

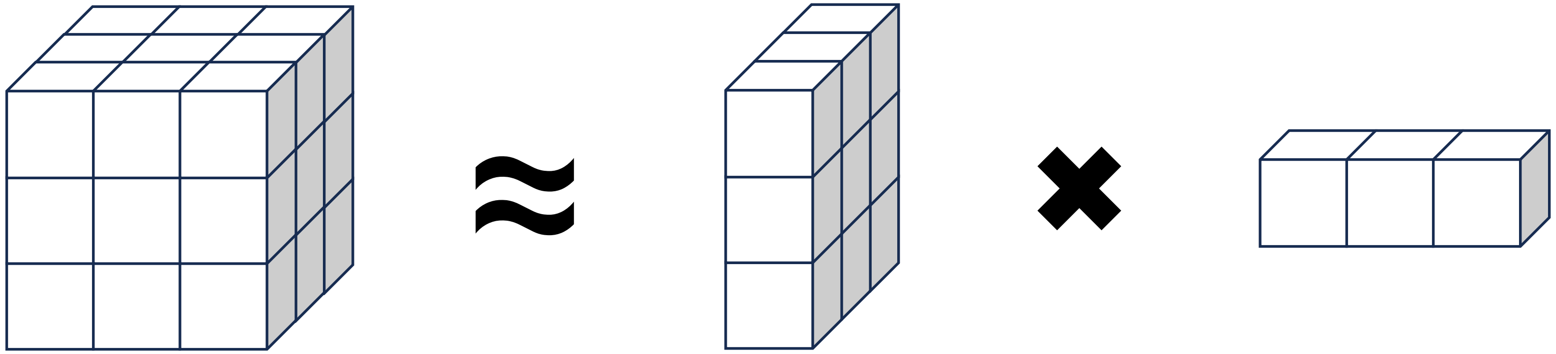
Early Fusion vs Late Fusion vs 3D CNN



Pseudo-3D CNNs

C3D Problem: 3x3x3 conv is very expensive.

Idea: replace 3D conv through 2D (spatial) followed by 1D (temporal)



$3 \times 3 \times 3$ $\approx$ $1 \times 3 \times 3$ $\times$ $3 \times 1 \times 1$

full 3x3x3 kernel

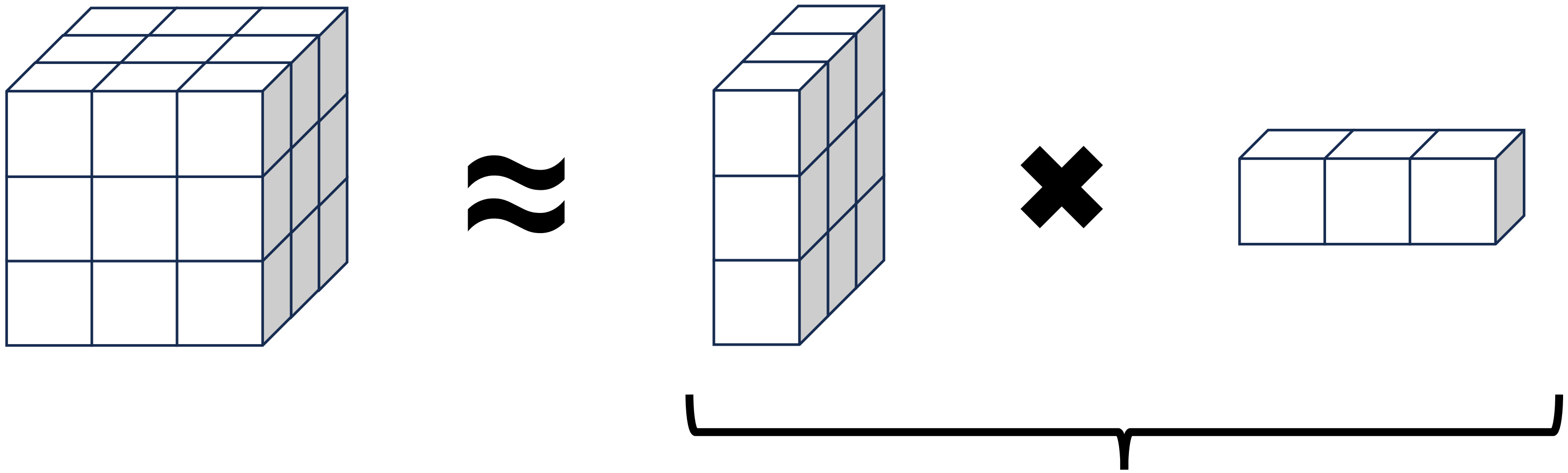
can be viewed as 3x3x3 kernel with shared weights along 1st dim

can be viewed as 3x3x3 kernel with shared weights along 2nd +3rd dim

Pseudo-3D CNNs

C3D Problem: 3x3x3 conv is very expensive.

Idea: replace 3D conv through 2D (spatial) followed by 1D (temporal)



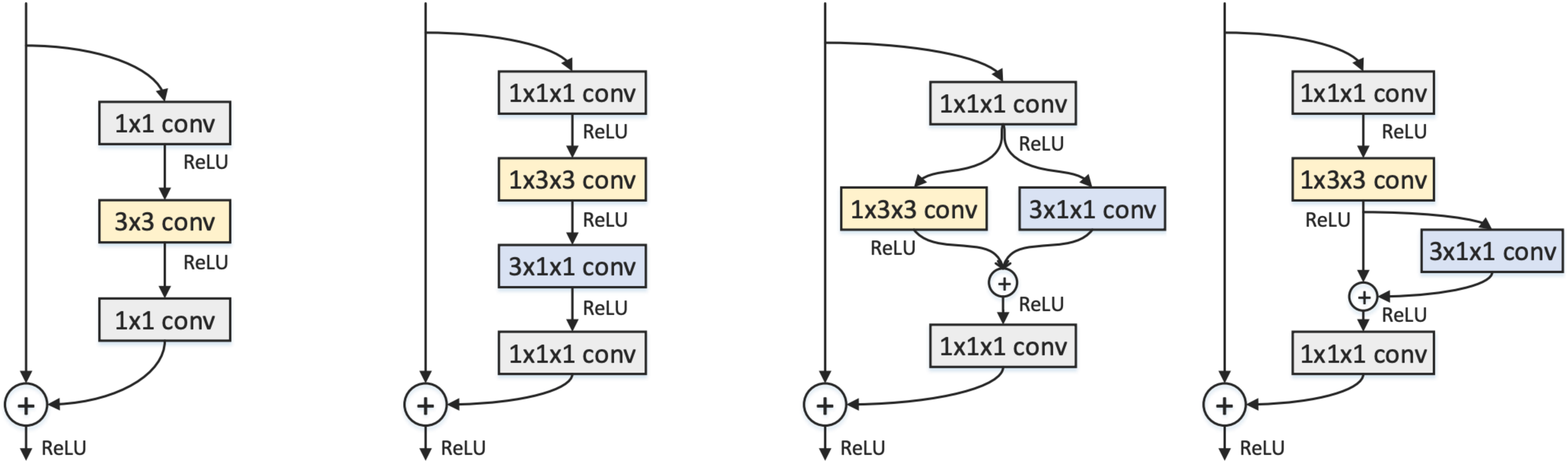
full 3x3x3 kernel

3x3x3 = **27xC params**

can be viewed as **structured** 3x3x3 kernel

3x3 + 3 = **11xC params**

Pseudo-3D CNNs



(a) Residual Unit [7] (b) P3D-A (c) P3D-B (d) P3D-C

Qiu et.al, "Learning Spatio-Temporal Representation with Pseudo-3D Residual Networks". ICCV 2017

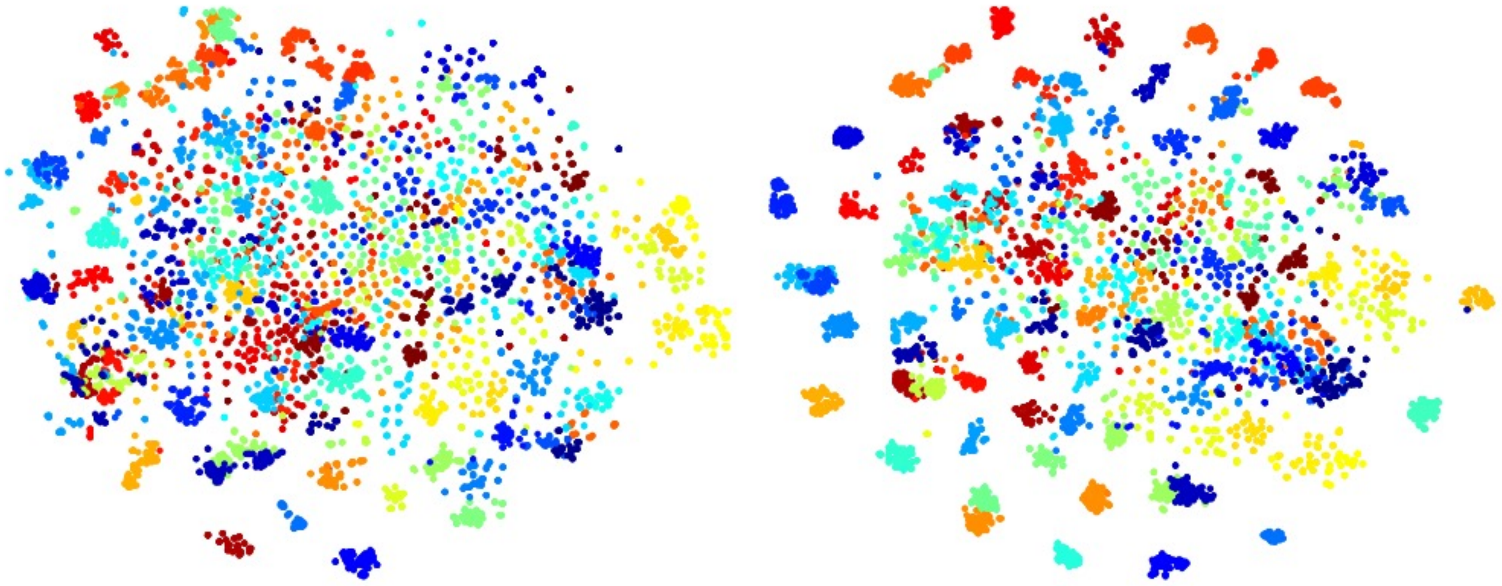
Pseudo-3D CNNs

Method	Pre-train Data	Clip Length	Clip hit@1	Video hit@1	Video hit@5
Deep Video (Single Frame) [10]	ImageNet1K	1	41.1%	59.3%	77.7%
Deep Video (Slow Fusion) [10]	ImageNet1K	10	41.9%	60.9%	80.2%
Convolutional Pooling [37]	ImageNet1K	120	70.8%	72.3%	90.8%
C3D [31]	–	16	44.9%	60.0%	84.4%
C3D [31]	I380K	16	46.1%	61.1%	85.2%
ResNet-152 [7]	ImageNet1K	1	46.5%	64.6%	86.4%
P3D ResNet (ours)	ImageNet1K	16	47.9%	66.4%	87.4%

on Sports-1M dataset

Qiu et.al, “Learning Spatio-Temporal Representation with Pseudo-3D Residual Networks”. ICCV 2017

Pseudo-3D CNNs



(a) ResNet-152 (b) P3D ResNet

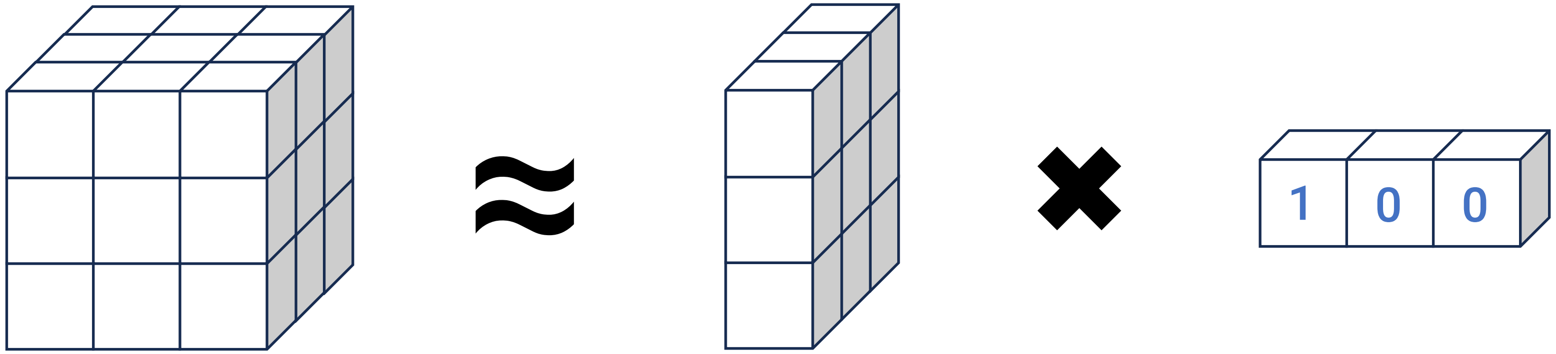
Figure 7. Video representation embedding visualizations of ResNet-152 and P3D ResNet on UCF101 using t-SNE [32]. Each video is visualized as one point and colors denote different actions.

Qiu et.al, “Learning Spatio-Temporal Representation with Pseudo-3D Residual Networks”. ICCV 2017

Pseudo-3D CNNs

C3D Problem: 3x3x3 conv is very expensive.

Idea: replace 3D conv through 2D (spatial) followed by 1D (temporal)

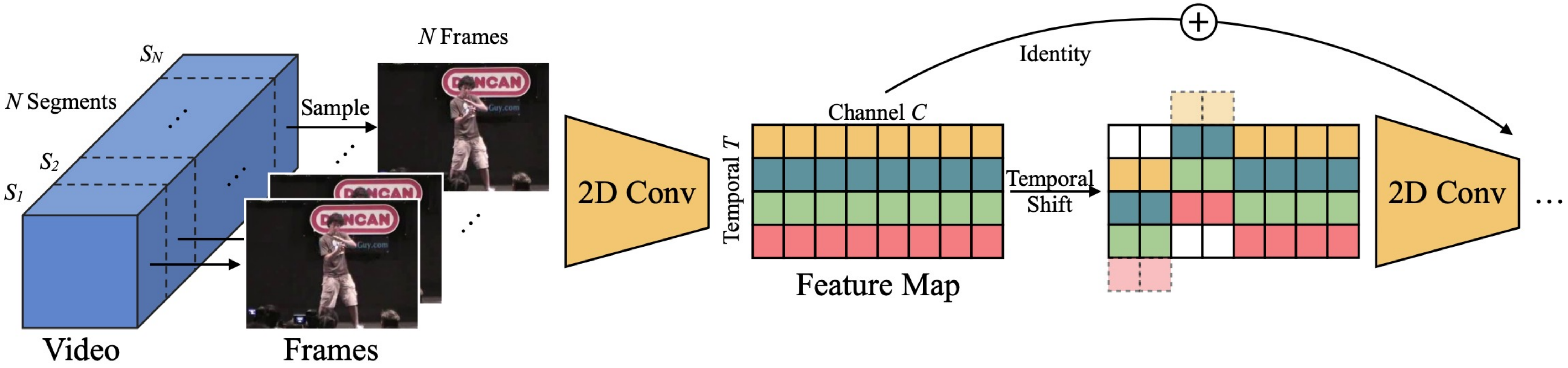


What if 1D temporal kernel is not learnt but hard-coded to [1,0,0] ?

TSM: Temporal Shift Module

Goal: Achieve 3D CNN performance at 2D CNN complexity

Idea: at each 2D CNN layer, shift part of the channels along the temporal dimension



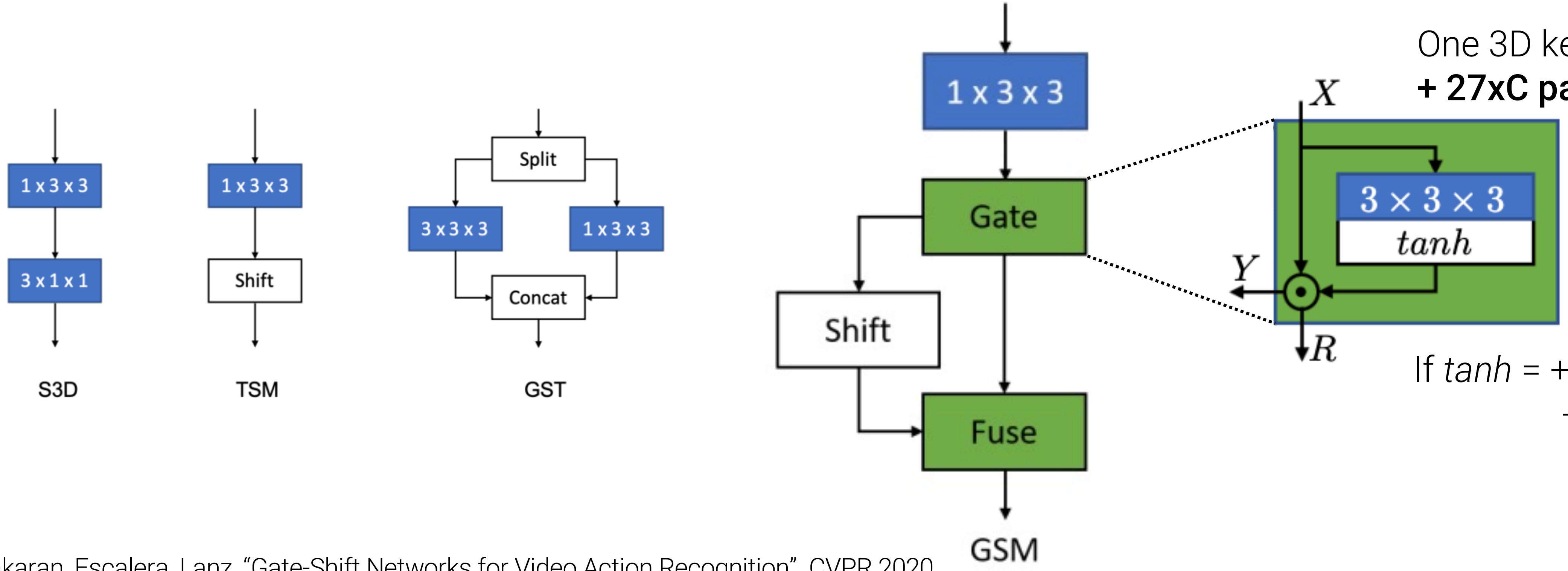
Shift is zero-flops, zero-params (but not zero-latency: in-memory data movement)

Qiu et.al, "Learning Spatio-Temporal Representation with Pseudo-3D Residual Networks". ICCV 2017

Gate-Shift-Fuse Networks

TSM shifts feature planes forward and backward in time. But not all feature regions may need to be shifted for improving action recognition performance.

Idea: add a learnable gate to decide which regions to shift, and which to keep

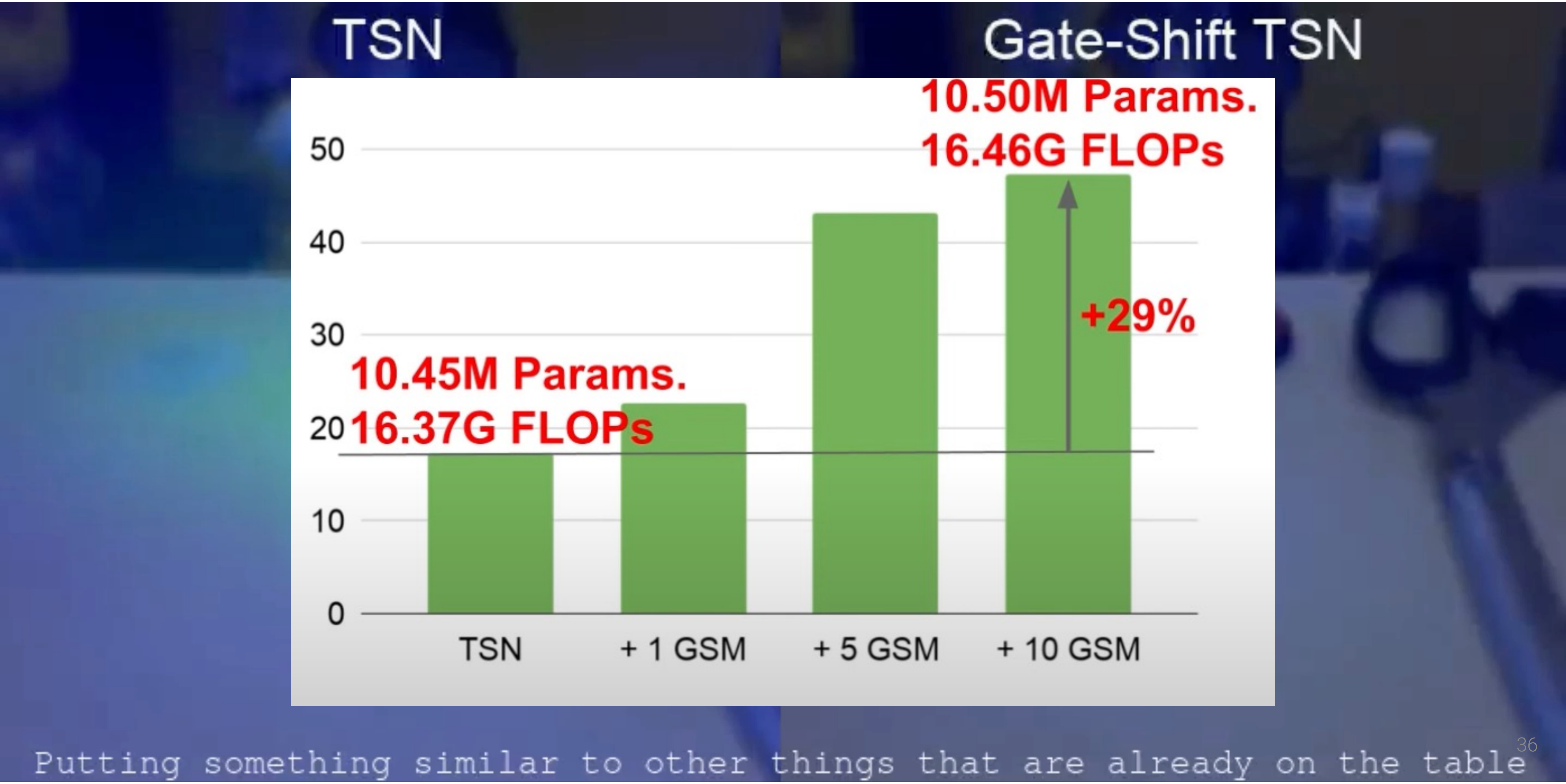


One 3D kernel, that is
+ 27xC params/layer

If *tanh* = +1: feat. mean
-1: feat. diff

Sudhakaran, Escalera, Lanz, "Gate-Shift Networks for Video Action Recognition". CVPR 2020
Sudhakaran, Escalera, Lanz, "Gate-Shift-Fuse for Video Action Recognition". TPAMI 2023

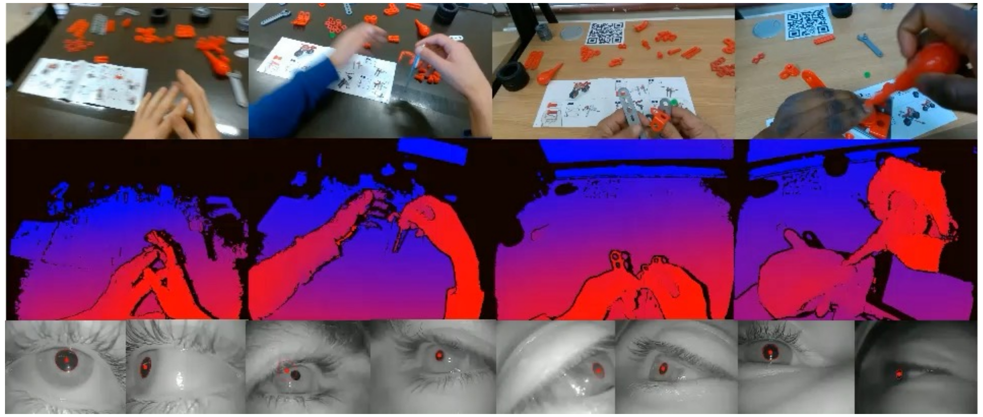
Gate-Shift-Fuse Networks



Gate-Shift-Fuse Networks

EPIC-Kitchens-100 dataset

Method	Backbone	Accuracy (%)		
		Verb	Noun	Action
TSN [62]*	ResNet-50	60.18	46.03	33.19
TRN [79]*	ResNet-50	65.88	45.43	35.34
TSM [37]*	ResNet-50	67.86	49.01	38.27
SlowFast [13]*	ResNet-50	65.56	50.02	38.54
MoViNet-A6 [27]	-	72.2	57.3	47.7
GSF	InceptionV3	68.35	52.71	43.42
	ResNet-50	68.76	52.74	44.04
	ResNet-101	69.97	54.01	44.78



MECCANO dataset

Modality	SlowFast		GSF		SlowFast GSF	
	Top1 (%)	Top5 (%)	Top1 (%)	Top5 (%)	Top1 (%)	Top5 (%)
RGB	45.16	73.75	45.09	75.47	**49.06**	**78.73**
Depth	45.13	72.19	45.44	75.54	**46.51**	**77.35**
RGB-Depth	49.49	77.61	50.30	79.19	**51.54**	**80.79**

Recognizing Actions from Motion

Johansson, "Visual Perception of Biological Motion and a Model for its Analysis. Perception & Psychophysics, 1973.

Motion representation: Optical Flow

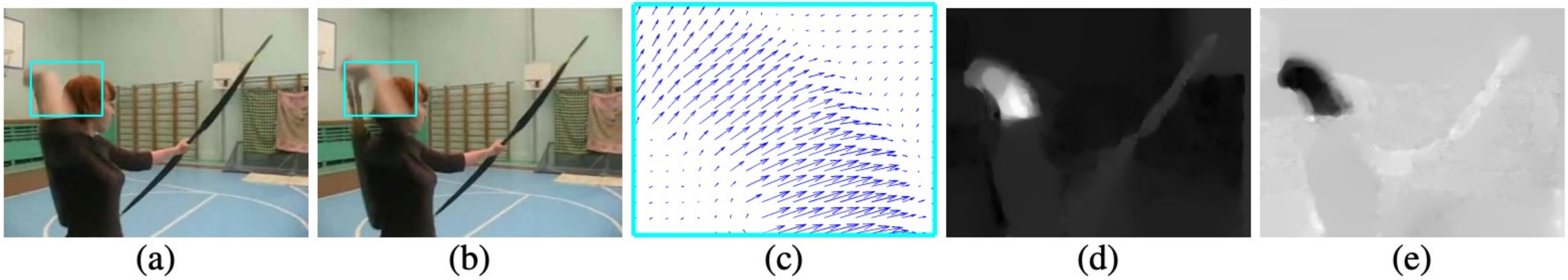
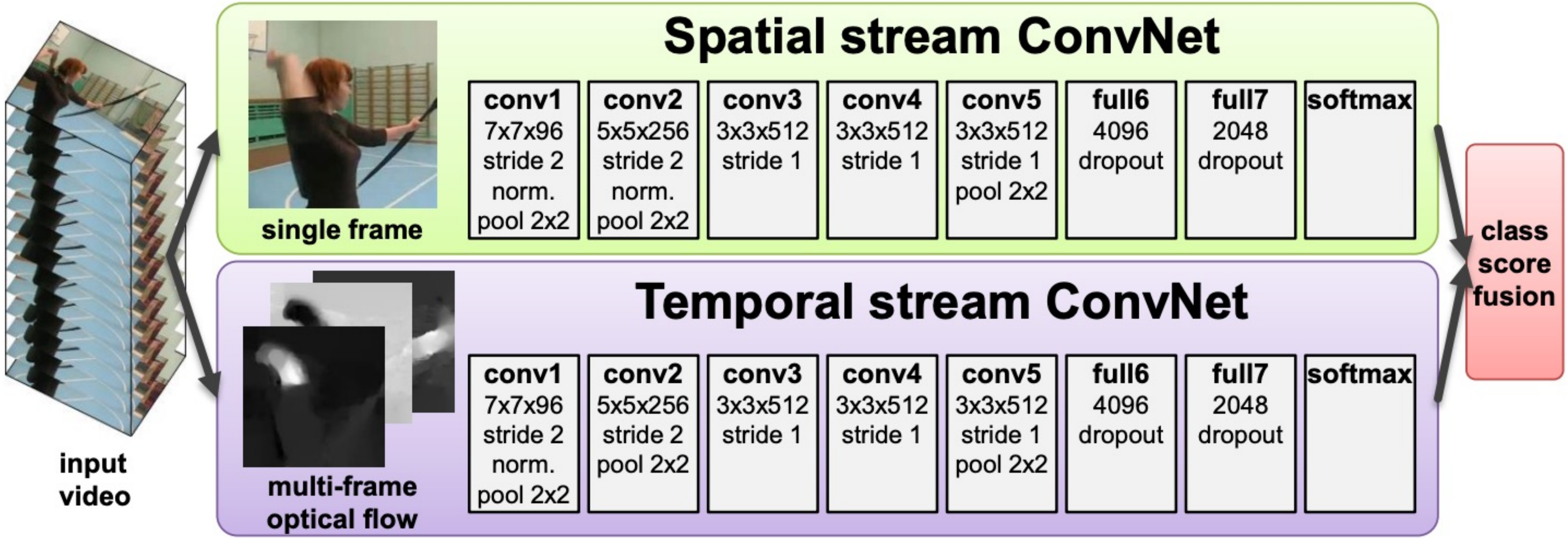


Figure 2: **Optical flow.** (a),(b): a pair of consecutive video frames with the area around a moving hand outlined with a cyan rectangle. (c): a close-up of dense optical flow in the outlined area; (d): horizontal component d^x of the displacement vector field (higher intensity corresponds to positive values, lower intensity to negative values). (e): vertical component d^y. Note how (d) and (e) highlight the moving hand and bow. The input to a ConvNet contains multiple flows (Sect. 3.1).

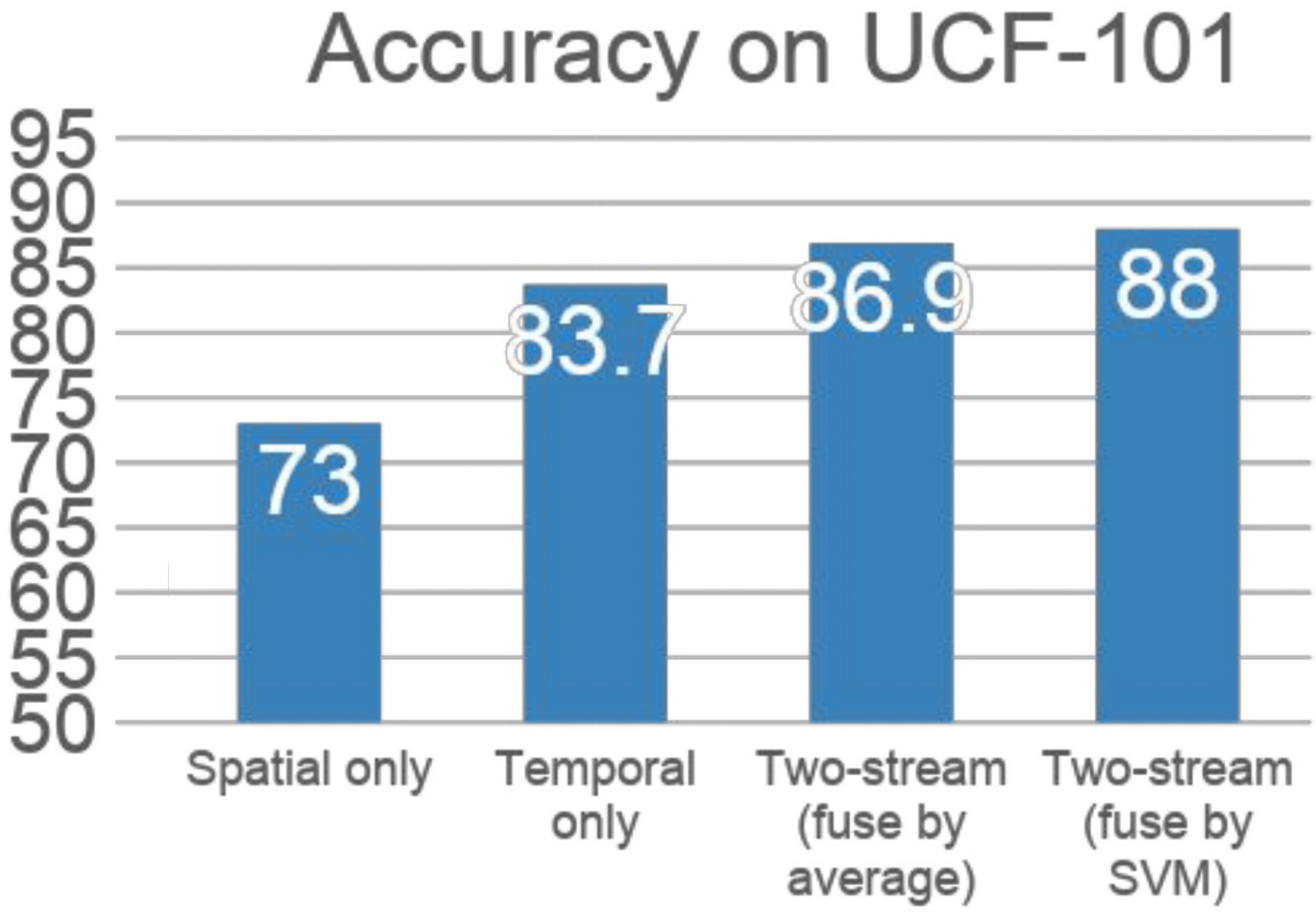
Simonyan and Zisserman: Two-Stream Convolutional Networks for Action Recognition in Videos. NIPS, 2014.

Two-Stream CNN



Simonyan and Zisserman: Two-Stream Convolutional Networks for Action Recognition in Videos. NIPS, 2014.

Separating Motion and Appearance: Two-Stream Networks

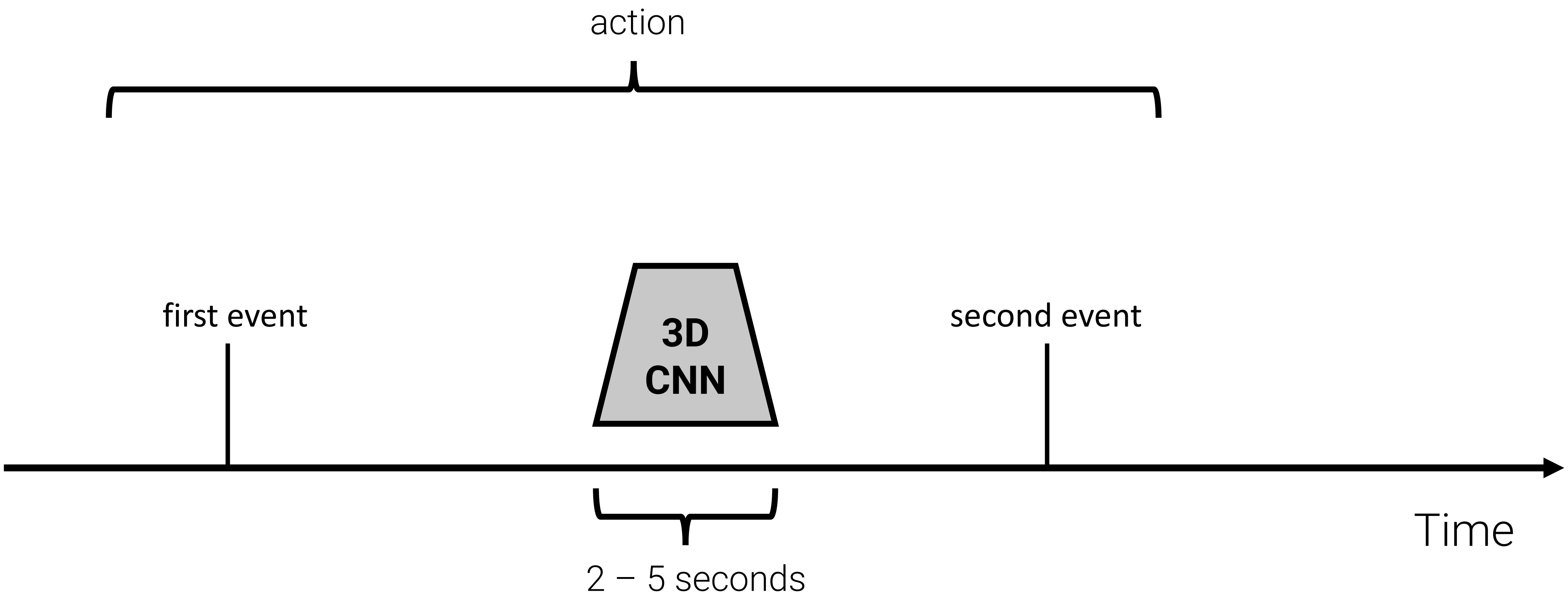


Simonyan and Zisserman: Two-Stream Convolutional Networks for Action Recognition in Videos. NIPS, 2014.

Modeling Long-Term Temporal Structure

So far, all our temporal CNNs only model local motion between frames in very short clips.

What about long-term structure ?



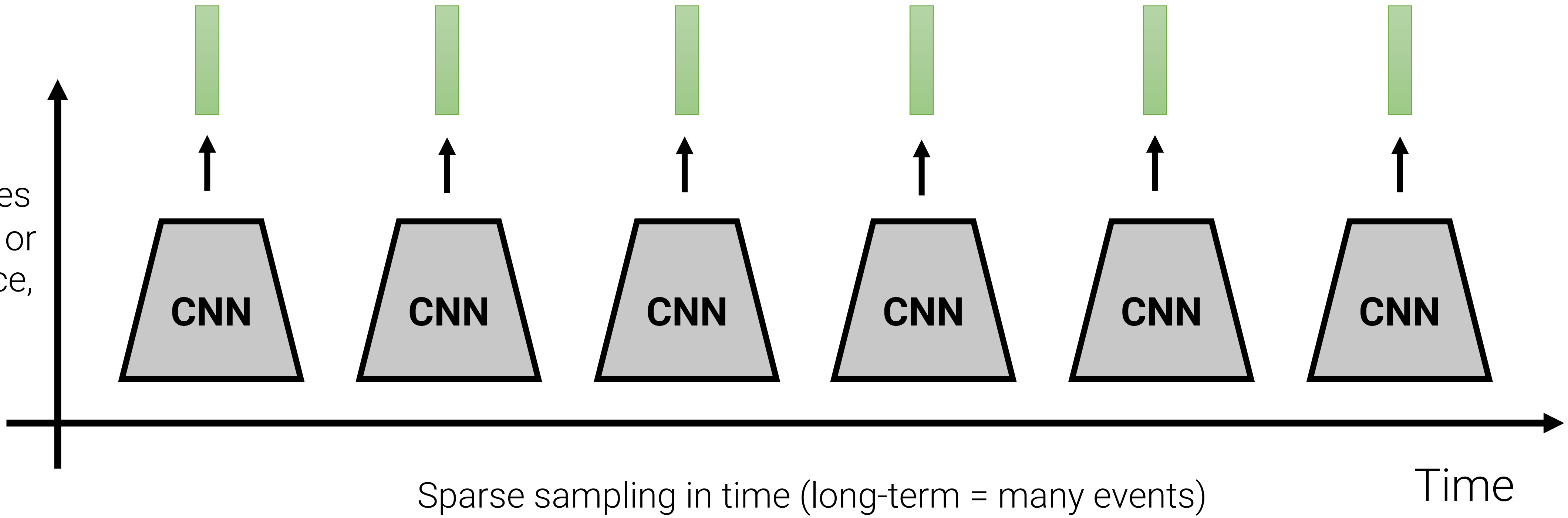
Modeling Long-Term Temporal Structure

Long-term video represented as sequence of features

How to aggregate the features to capture temporal structure?

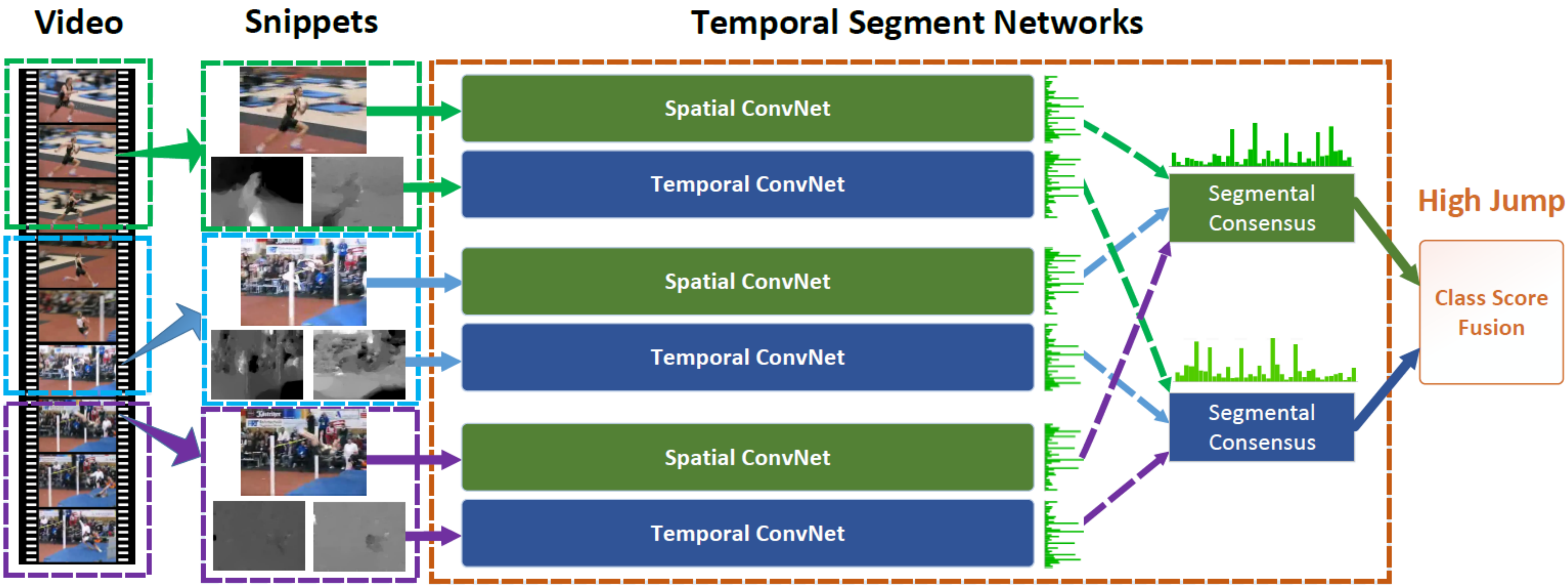
Note that AvgPool over time (as late fusion) would yield invariance to frame reshuffling

Extract features with CNN (2D or 3D, appearance, flow)



Sparse sampling in time (long-term = many events)

Temporal Segment Networks

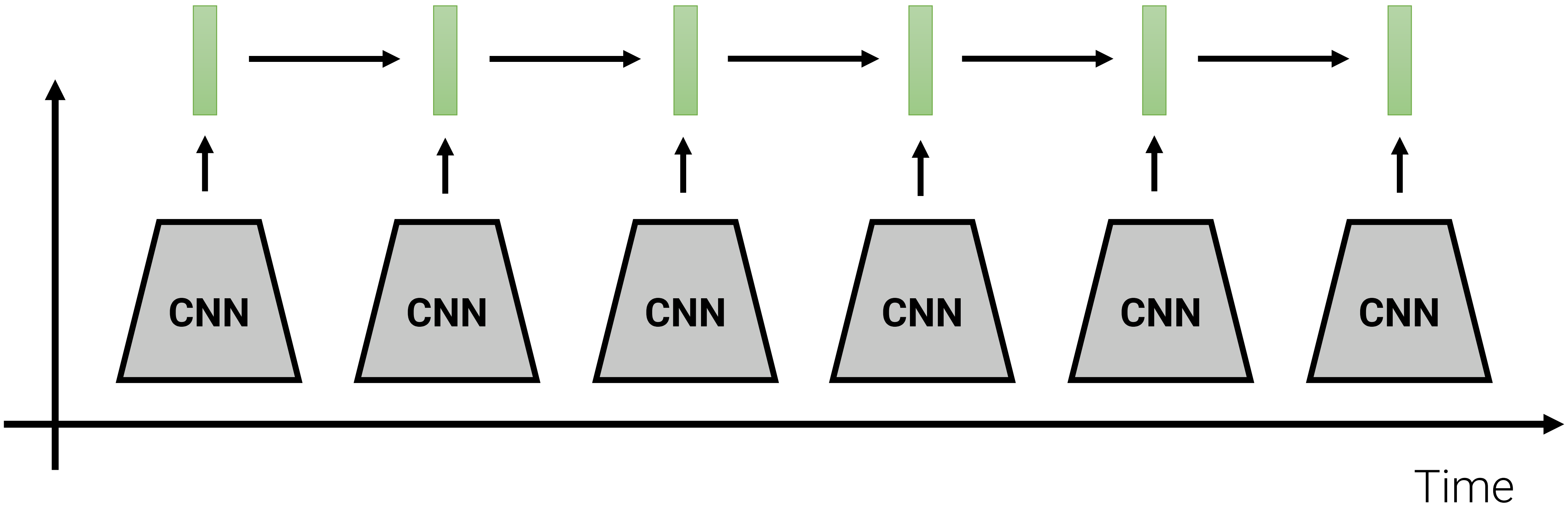


Wang et.al: Temporal Segment Networks: Towards Good Practices for Deep Action Recognition., CVPR, 2016.

Modeling Long-Term Temporal Structure

Process local features using recurrent networks (e.g., LSTM)

- Inside CNN: each value is a function of fixed temporal window (local temporal structure)
- Inside RNN: each vector is a function of all previous vectors (global temporal structure)



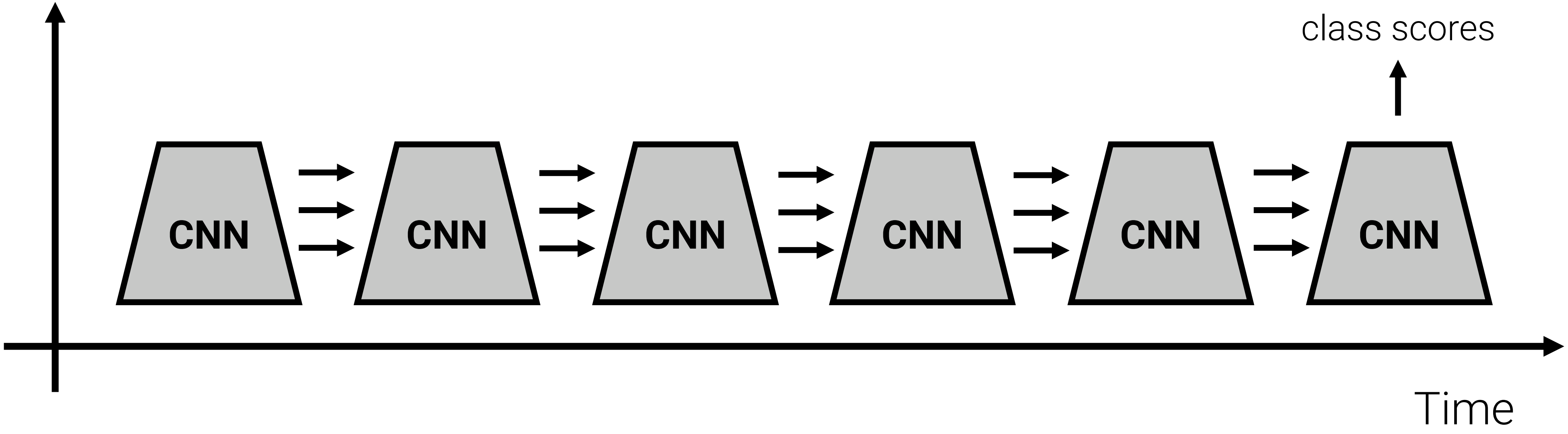
Donahue et.al, “Long-term recurrent convolutional networks for visual recognition and description”, CVPR 2015

Modeling Long-Term Temporal Structure

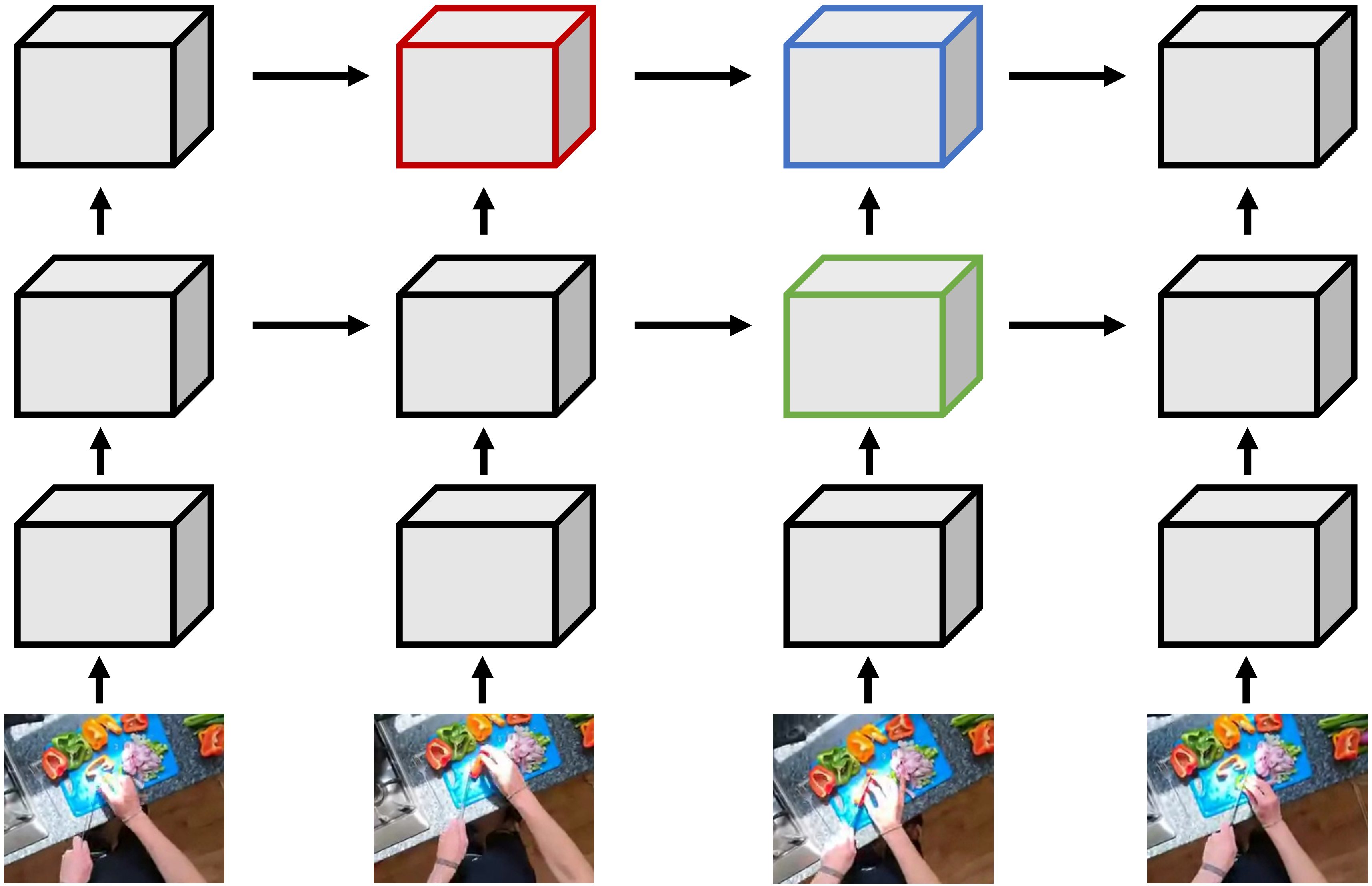
Process local features using recurrent networks (e.g., LSTM)

- Inside CNN: each value is a function of fixed temporal window (local temporal structure)
- Inside RNN: each vector is a function of all previous vectors (global temporal structure)

Can we merge both approaches, i.e. go deep with recurrence ?



Recurrent Convolutional Network



Entire network uses 2D feature maps

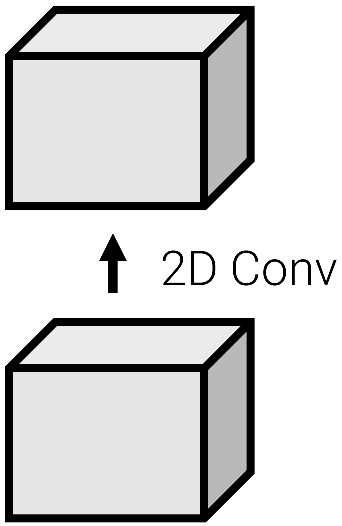
Each depends on two inputs:
- same layer, previous input
- previous layer, same timestep

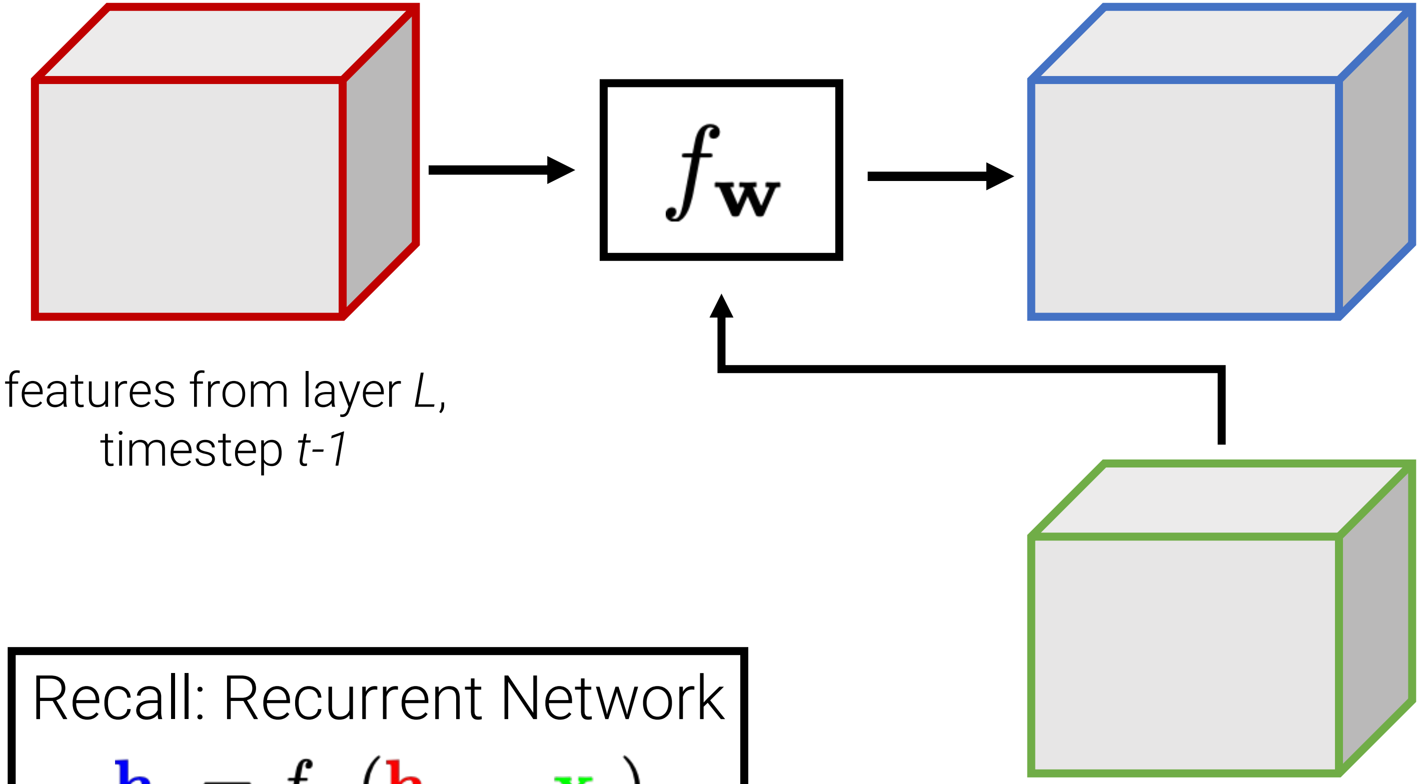
As in multi-layer RNN
- different weights at each layer
- share weights across time

Ballas et al, "Delving Deeper into Convolutional Networks for Learning Video Representations", ICLR 2016

Recurrent Convolutional Network

Standard 2D CNN





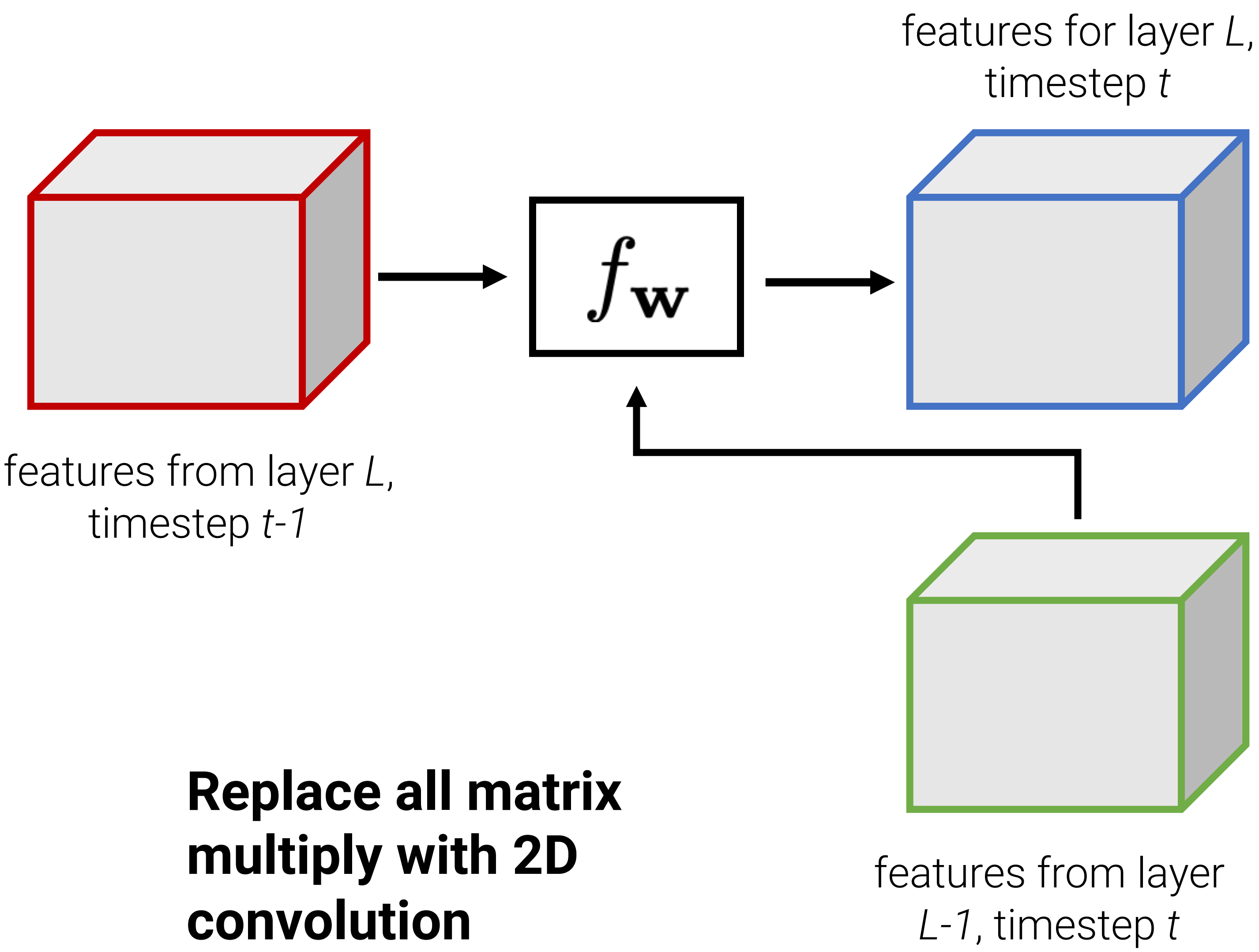
Recall: Recurrent Network

$$\mathbf{h}_t = f_{\mathbf{w}}(\mathbf{h}_{t-1}, \mathbf{x}_t)$$

some function with parameters **w**

Ballas et al, "Delving Deeper into Convolutional Networks for Learning Video Representations", ICLR 2016

Recurrent Convolutional Network



Vanilla RNN:

$$\mathbf{h}_t = \tanh(\mathbf{A}\mathbf{h}_{t-1} + \mathbf{B}\mathbf{x}_t)$$

GRU:

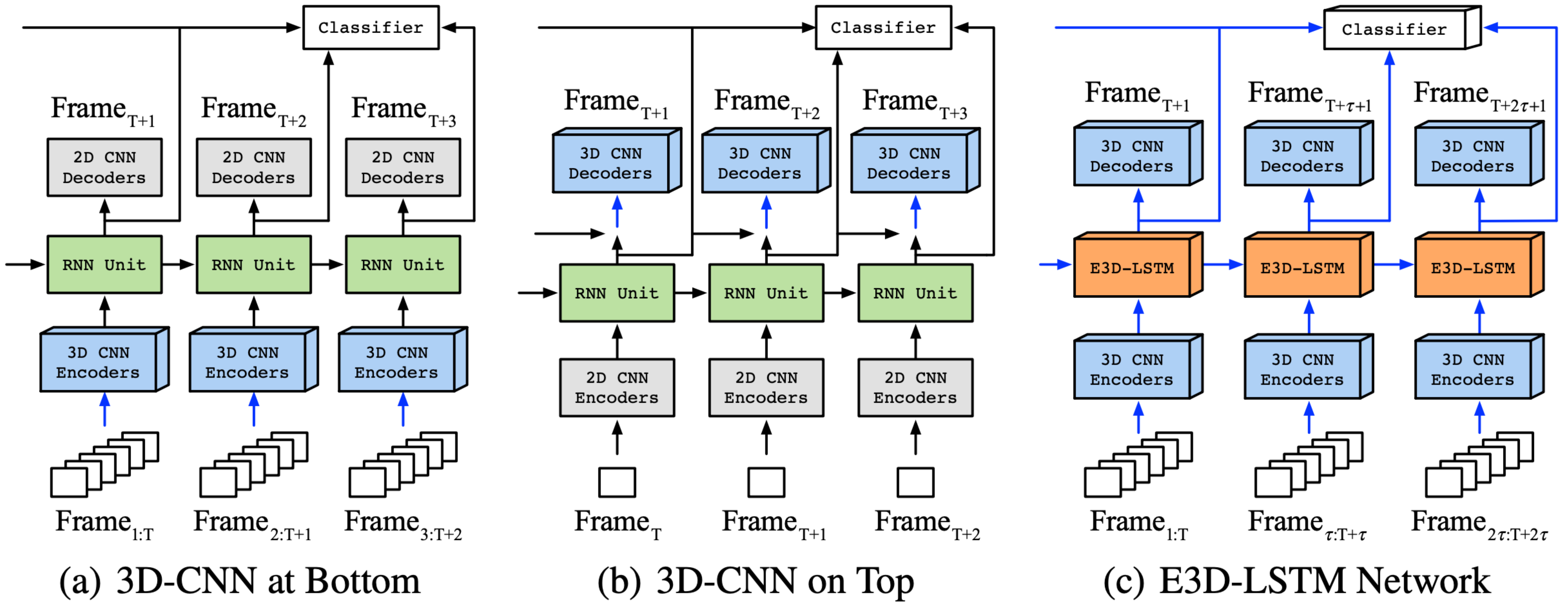
$$\begin{aligned}
\mathbf{r}_t &= \mathrm{sigm}(\mathbf{A}_r\mathbf{h}_{t-1} + \mathbf{B}_r\mathbf{x}_t) \\
\mathbf{u}_t &= \mathrm{sigm}(\mathbf{A}_u\mathbf{h}_{t-1} + \mathbf{B}_u\mathbf{x}_t) \\
\mathbf{s}_t &= \tanh(\mathbf{A}_s(\mathbf{r}_t \odot \mathbf{h}_{t-1}) + \mathbf{B}_s\mathbf{x}_t) \\
\mathbf{h}_t &= \mathbf{u}_t \odot \mathbf{h}_{t-1} + (1 - \mathbf{u}_t) \odot \mathbf{s}_t
\end{aligned}$$

LSTM:

...

Replace all matrix multiply with 2D convolution

Eidetic 3D LSTM



Wang et al, “Eidetic 3D LSTM: A Model for Video Prediction and Beyond”, ICLR 2019

Eidetic 3D LSTM

Table 2: Ablation study on the Moving MNIST dataset ($10 \rightarrow 10$).

MODEL	SSIM	MSE
BASELINE 1: 3D-CNN AT BOTTOM (FIGURE 1(A))	0.859	50.6
BASELINE 2: 3D-CNN ON TOP (FIGURE 1(B))	0.862	53.4
BASELINE 3: OURS (W/O 3D CONVOLUTIONS)	0.894	44.2
BASELINE 4: OURS (W/O MEMORY ATTENTION)	0.880	45.7
E3D-LSTM	**0.910**	**41.3**

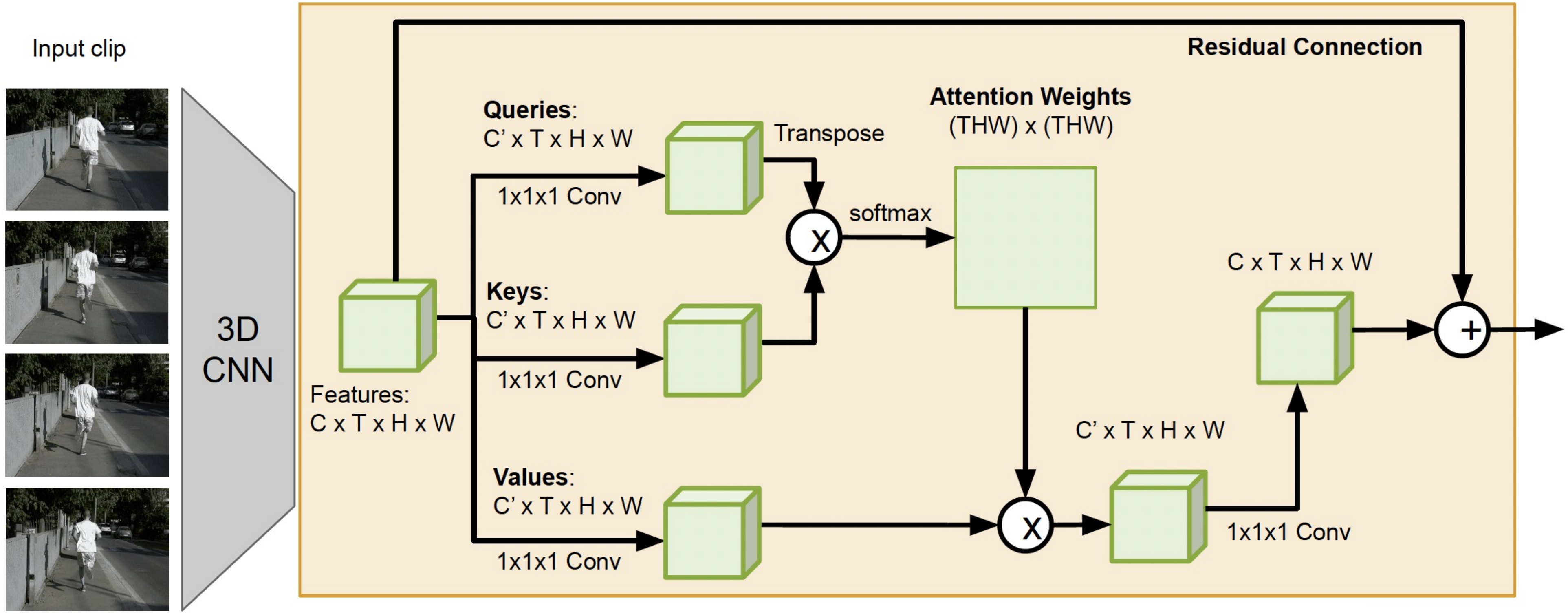
Table 5: Early activity recognition accuracy on the 41-category subset of Something-Something.

MODEL	FRONT 25%	FRONT 50%
3D-CNN	9.11	10.30
SEPARABLE-CNN: SEPARABLE-CONV AT BOTTOM	8.94	9.62
(2+1)D-CNN: SEPARABLE-CONV ON TOP	9.08	10.17
E(2+1)D-LSTM: SEPARABLE INSIDE UNITS	12.45	19.86
E3D-LSTM	**14.59**	**22.73**

Wang et al, "Eidetic 3D LSTM: A Model for Video Prediction and Beyond", ICLR 2019

Modeling Long-Term Temporal Structure

Problem: RNNs are slow for long sequences (can't be parallelized)



Spatio-temporal self-attention (Non-local Block)

We can add non-local blocks into existing 3D CNNs (at multiple layers)

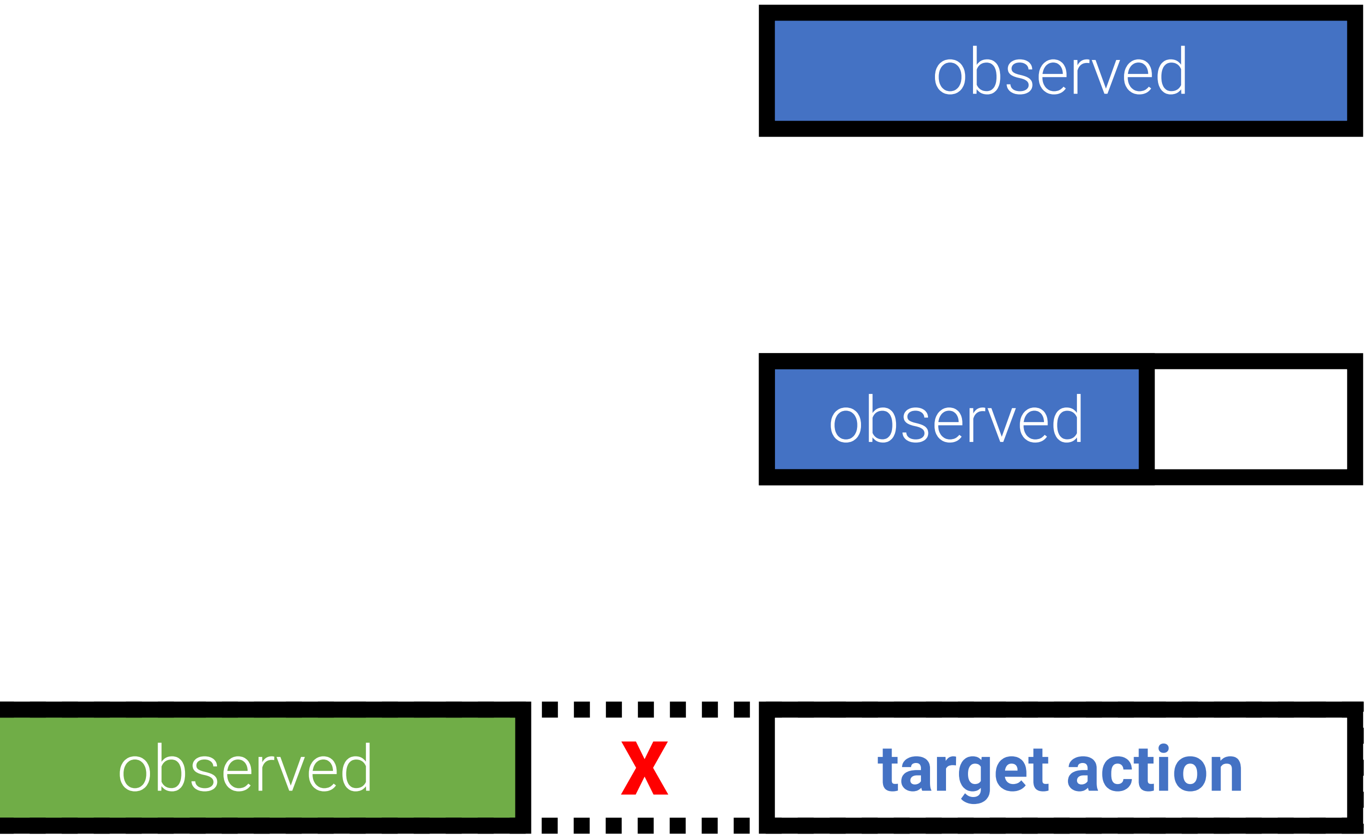
Wang et al, "Non-local Neural Networks", CVPR 2018

Action Prediction Models

Introducing

Ego4D

Early Action Recognition vs Action Anticipation



Action Recognition (= Trimmed Video Classification with Action Labels)

Early Action Recognition

Action Anticipation/Prediction

Early Action Recognition vs Action Anticipation

observed

Action Recognition (= Trimmed Video Classification with Action Labels)

Challenges: intra-class variations, clutter, viewpoint, occlusion, dynamic background, camera motion (ego-centric), sensor noise & synchronization (multimodal) …

Early Action Recognition vs Action Anticipation

Action Recognition (= Trimmed Video Classification with Action Labels)

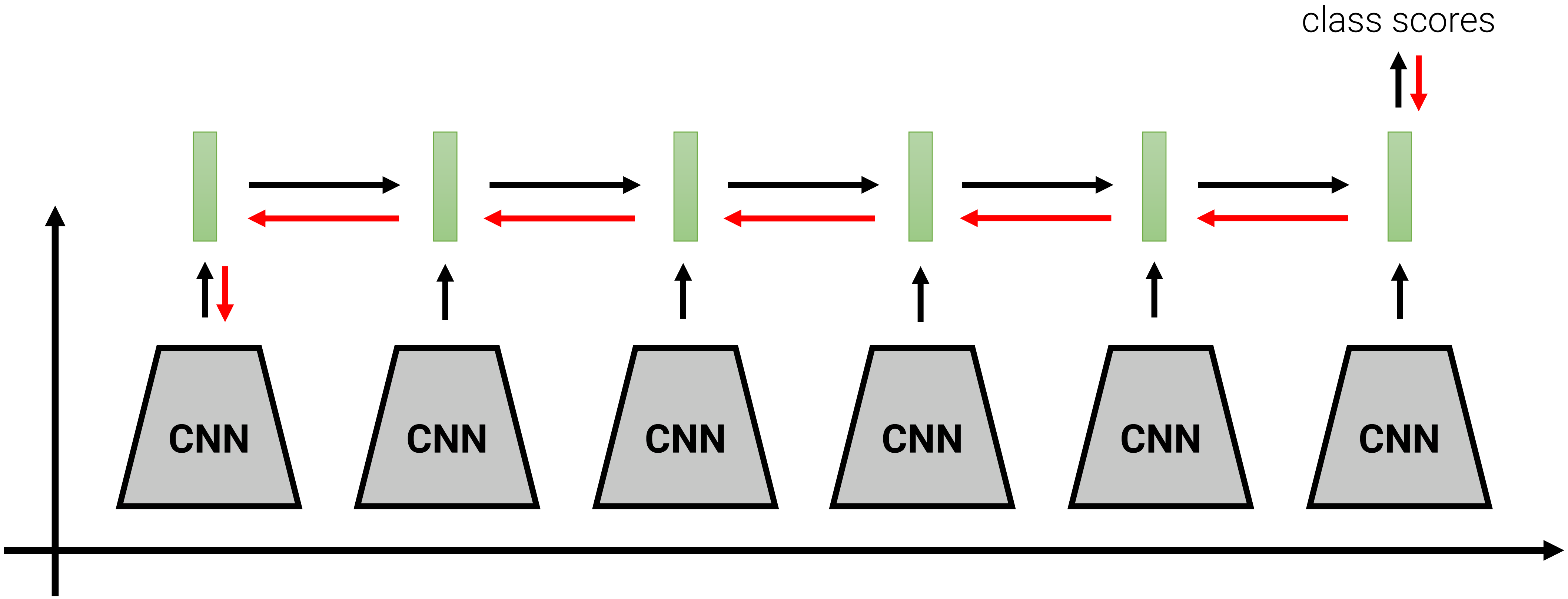
Early Action Recognition

+ incomplete observation (only initial part of action is observed, remaining part is fully occluded)

Improving Gradient Flow

Recall: Vanishing Gradients prevent effective learning of long range dependencies

$$\frac{\partial \mathbf{h}_t}{\partial \mathbf{h}_{t-k}} = \frac{\partial \mathbf{h}_t}{\partial \mathbf{h}_{t-1}} \frac{\partial \mathbf{h}_{t-1}}{\partial \mathbf{h}_{t-2}} \frac{\partial \mathbf{h}_{t-2}}{\partial \mathbf{h}_{t-3}} \cdots \frac{\partial \mathbf{h}_{t-k+1}}{\partial \mathbf{h}_{t-k}}$$



Improving Gradient Flow

Recall: Vanishing Gradients prevent effective learning of long range dependencies

$$\frac{\partial \mathbf{h}_t}{\partial \mathbf{h}_{t-k}} = \frac{\partial \mathbf{h}_t}{\partial \mathbf{h}_{t-1}} \frac{\partial \mathbf{h}_{t-1}}{\partial \mathbf{h}_{t-2}} \frac{\partial \mathbf{h}_{t-2}}{\partial \mathbf{h}_{t-3}} \quad \cdots \quad \frac{\partial \mathbf{h}_{t-k+1}}{\partial \mathbf{h}_{t-k}}$$

$$\mathbf{h}_t = \tanh(\mathbf{A}\mathbf{h}_{t-1} + \mathbf{B}\mathbf{x}_t) \quad \Rightarrow \quad \frac{\partial \mathbf{h}_t}{\partial \mathbf{h}_{t-1}} \approx \mathbf{A} \quad \Rightarrow \quad \frac{\partial \mathbf{h}_t}{\partial \mathbf{h}_{t-k}} \approx \mathbf{A}^k = (\mathbf{Q}\boldsymbol{\Lambda}\mathbf{Q}^\top)^k = \mathbf{Q}\boldsymbol{\Lambda}^k\mathbf{Q}^\top$$

Vanilla RNN

Components with eigenvalues > 1: exploding gradients

Components with eigenvalues < 1: vanishing gradients

Improving Gradient Flow

Recall: Vanishing Gradients prevent effective learning of long range dependencies

$$\frac{\partial \mathbf{h}_t}{\partial \mathbf{h}_{t-k}} = \frac{\partial \mathbf{h}_t}{\partial \mathbf{h}_{t-1}} \frac{\partial \mathbf{h}_{t-1}}{\partial \mathbf{h}_{t-2}} \frac{\partial \mathbf{h}_{t-2}}{\partial \mathbf{h}_{t-3}} \quad \cdots \quad \frac{\partial \mathbf{h}_{t-k+1}}{\partial \mathbf{h}_{t-k}}$$

$$\mathbf{h}_t = \tanh(\mathbf{A}\mathbf{h}_{t-1} + \mathbf{B}\mathbf{x}_t) \quad \Rightarrow \quad \frac{\partial \mathbf{h}_t}{\partial \mathbf{h}_{t-1}} \approx \mathbf{A} \quad \Rightarrow \quad \frac{\partial \mathbf{h}_t}{\partial \mathbf{h}_{t-k}} \approx \mathbf{A}^k = (\mathbf{Q}\mathbf{\Lambda}\mathbf{Q}^\top)^k = \mathbf{Q}\mathbf{\Lambda}^k\mathbf{Q}^\top$$

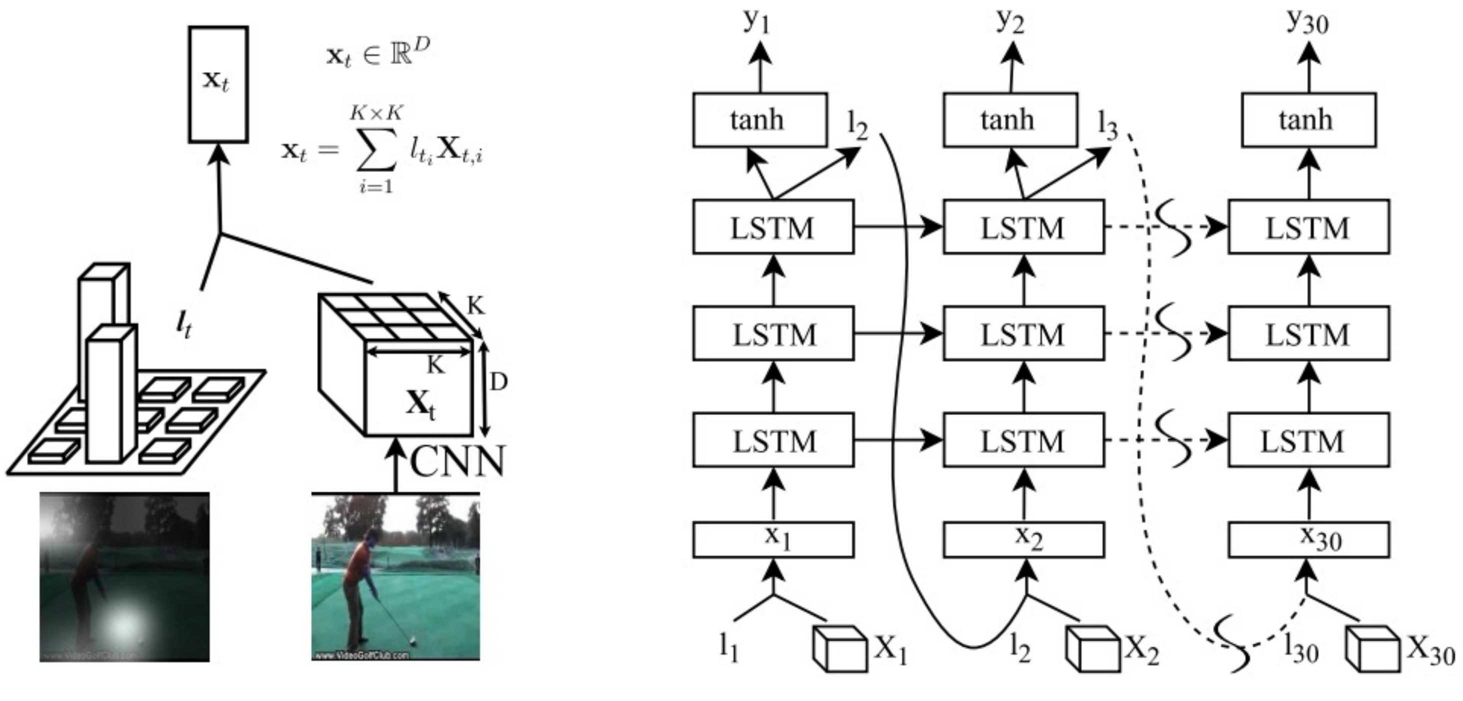
GRU, LSTM can maintain gradient flow despite small **A** by setting its gate to **u** ≈ 1

$$\mathbf{h}_t = \mathbf{u}_t \odot \mathbf{h}_{t-1} + (1 - \mathbf{u}_t) \odot \mathbf{s}_t \quad \Rightarrow \quad \frac{\partial \mathbf{h}_t}{\partial \mathbf{h}_{t-1}} = \frac{\partial \mathbf{u}_t}{\partial \mathbf{h}_{t-1}} \odot \mathbf{h}_{t-1} + \mathbf{u}_t + \ldots$$

u ≈ 1 like a skip connection

Improving Information Flow

Not all image/feature regions may be equally important => spatial attention



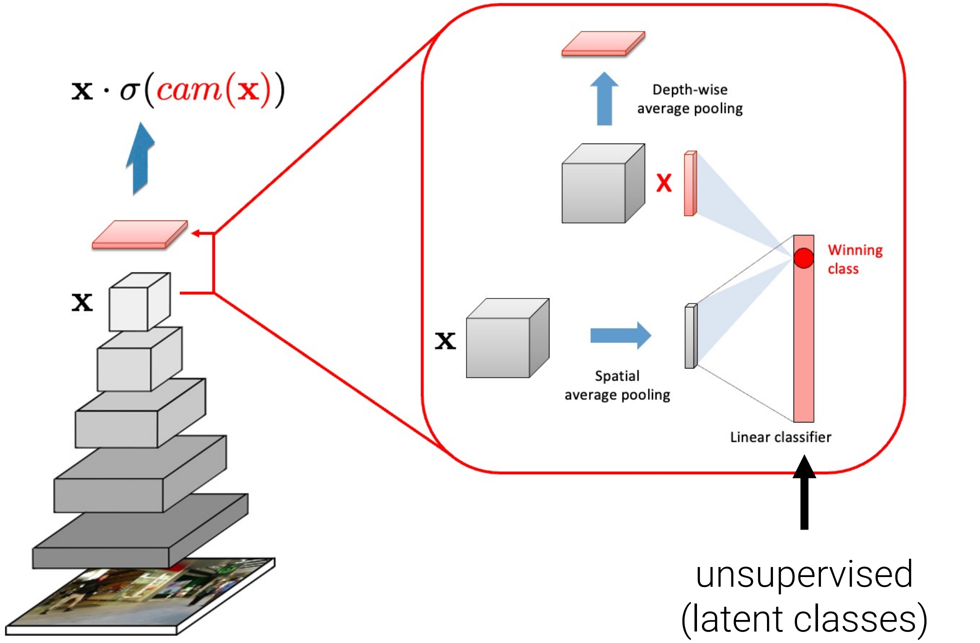
(a) The soft attention mechanism

(b) Our recurrent model

Sharma et al, "Action Recognition using Visual Attention", ICLR 2016

Long Short-Term Attention (LSTA)

Idea: build in spatial attention mechanisms into Convolutional LSTM cell



unsupervised (latent classes)

Class Activation Mapping (CAM) based spatial attention →

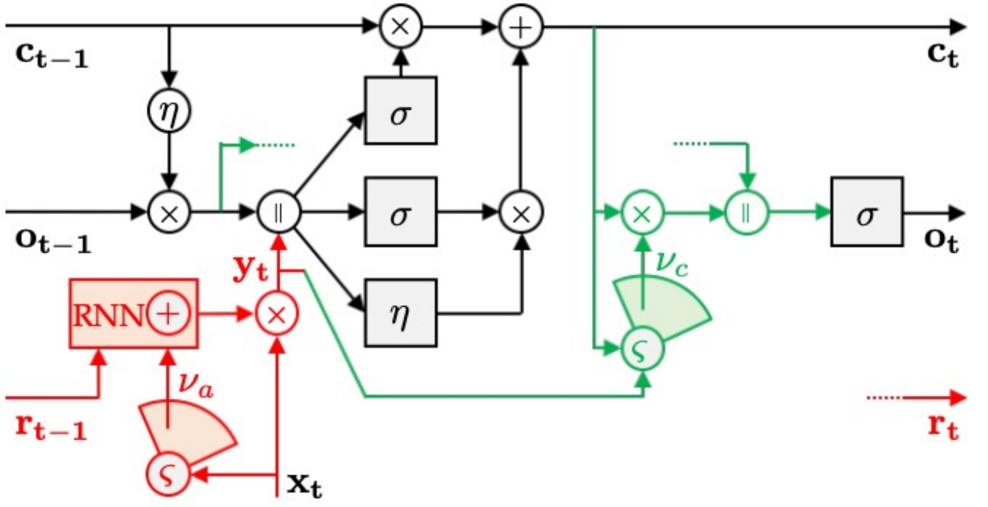
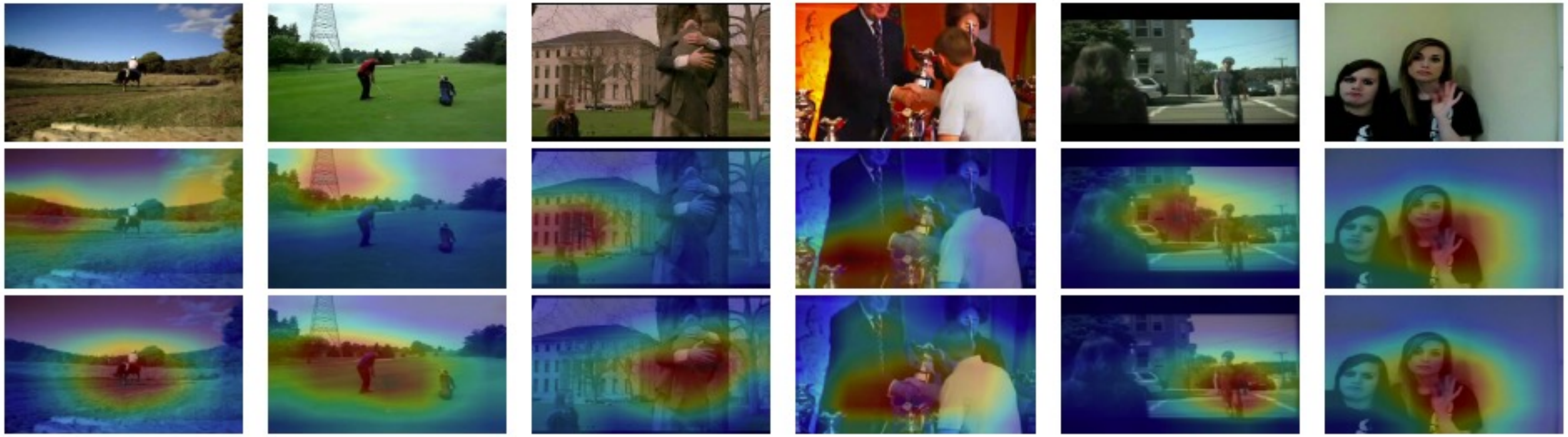


Fig. 2: LSTA extends LSTM (black part) with two novel components: recurrent attention and fine-grained output gating. The first (red part, *rca-attn* in Eq. (16)) tracks a weight map to focus on relevant feature regions, while the second (green part, Eq. (22)) introduces a high-capacity output gate. At the core of both is a spatial self-attention $\varsigma(\cdot, A)$ that pools parameters from attention dictionary A.

Sudhakaran, Escalera, Lanz, "LSTA: Long Short-Term Attention for Egocentric Action Recognition". CVPR 2019
Sudhakaran, Escalera, Lanz, "Learning to Recognize Actions on Objects in Egocentric Video With Attention Dictionaries". TPAMI 2023

CAM attention

Input
Prior
Learnt



(c) Ride horse (d) Golf (e) Hug (f) Shake hands (g) Walk (h) Wave

Method	Backbone	HMDB51		UCF101	
		RGB	RGB+Flow	RGB	RGB+Flow
Two-Stream VGG [4]	VGG-M	40.5	59.4	73.0	88
Two-Stream ResNet [6]	ResNet-50	43.4	60.6	82.3	89.5
TDD [5]	VGG-M	50	63.2	82.8	90.3
I3D [9]	Inception V1	49.8	66.4	84.5	93.4
TSN [7]	Inception V2	51	68.5	85.1	94
LSTM Soft Attention [16]	GoogleNet	41.3	-	84.9	-
ActionVLAD [12]	VGG-16	49.8	66.9	80.3	92.7
TA-VLAD (ours)	ResNet-34	**55.1**	**68.7**	**85.7**	**95.3**

Sudhakaran and Lanz, "Top-down Attention Recurrent VLAD Encoding for Action Recognition in Video", AI*IA 2018

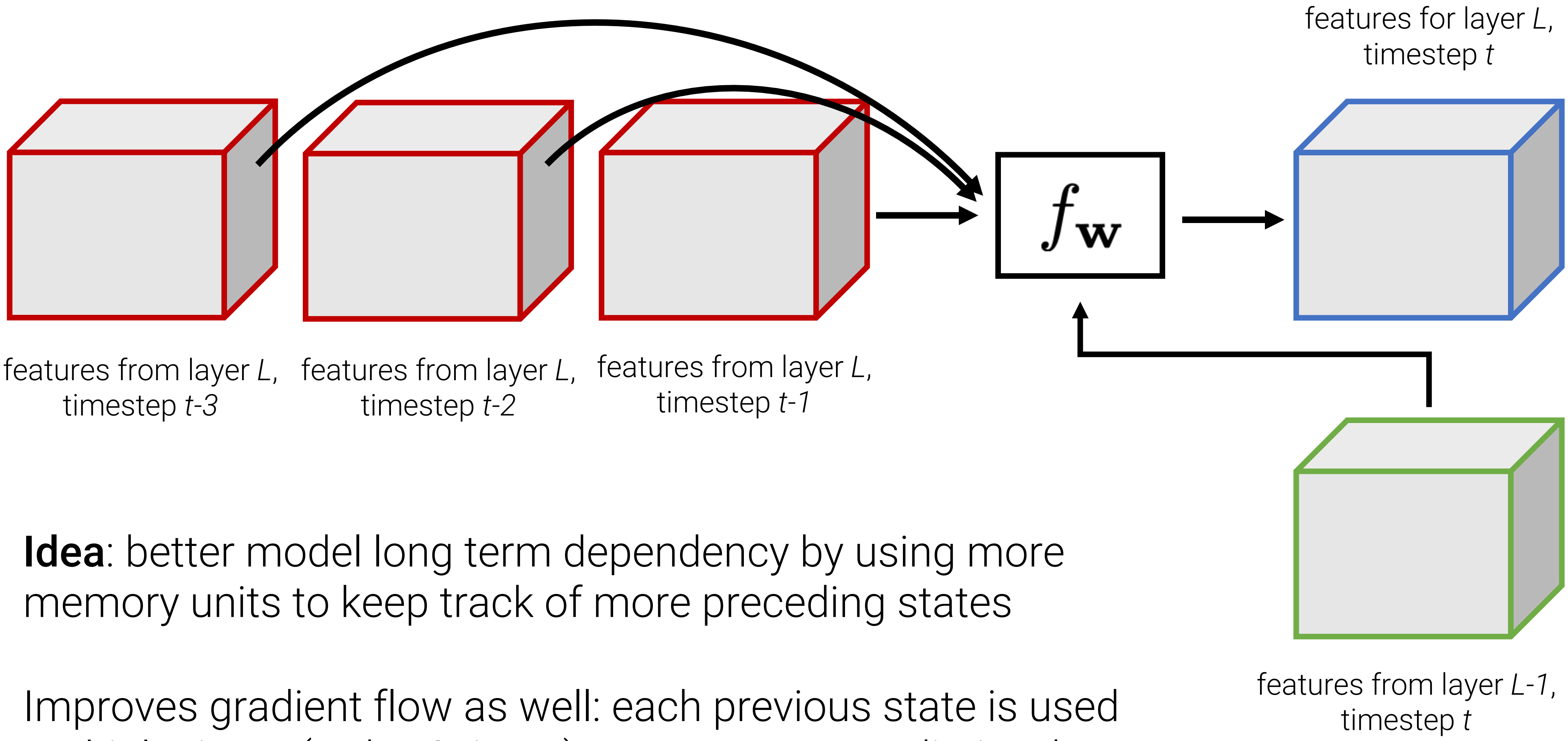
Long Short-Term Attention (LSTA)

Ablation on EPIC-Kitchens dataset

Method	Verb	Noun	Action
Baseline (ConvLSTM)	35.16/74.7	16/36.57	9.87/21.93
Baseline + rca-attn	39.14/73.89	16.95/38.19	12.25/25.62
Baseline + fine-grained output gating	46/76.94	21.32/41.73	13.75/28.71
Baseline + rca-attn + fine-grained output gating	45.81/77.47	22.36/45.16	14.92/30.43
LSTA	47.21/78.38	22.19/45.65	15.09/30.79

Sudhakaran, Escalera, Lanz, "LSTA: Long Short-Term Attention for Egocentric Action Recognition". CVPR 2019
Sudhakaran, Escalera, Lanz, "Learning to Recognize Actions on Objects in Egocentric Video With Attention Dictionaries". TPAMI 2023

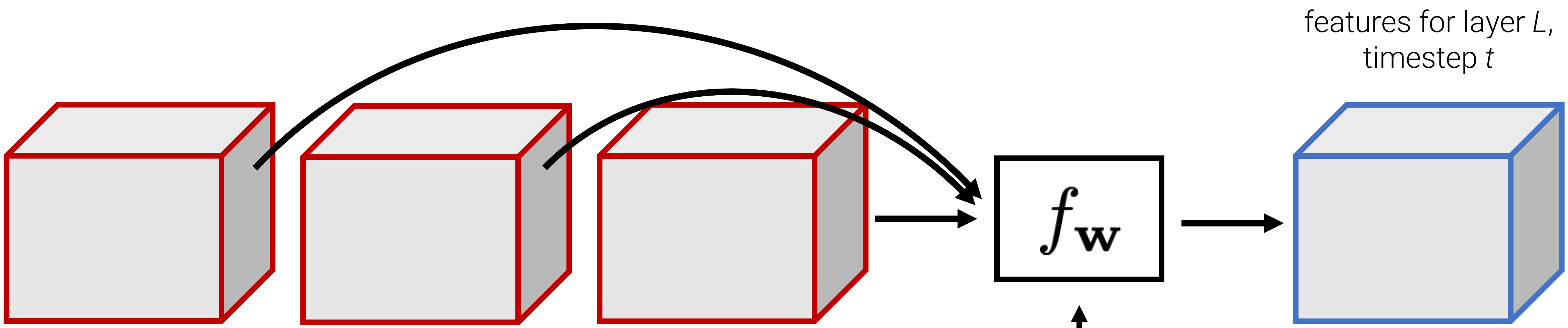
Higher Order Recurrent Convolutional Network



Idea: better model long term dependency by using more memory units to keep track of more preceding states

Improves gradient flow as well: each previous state is used multiple times (order-*S* times) to compute a prediction, hence gradient at a node accumulates *S* contributions during backprop

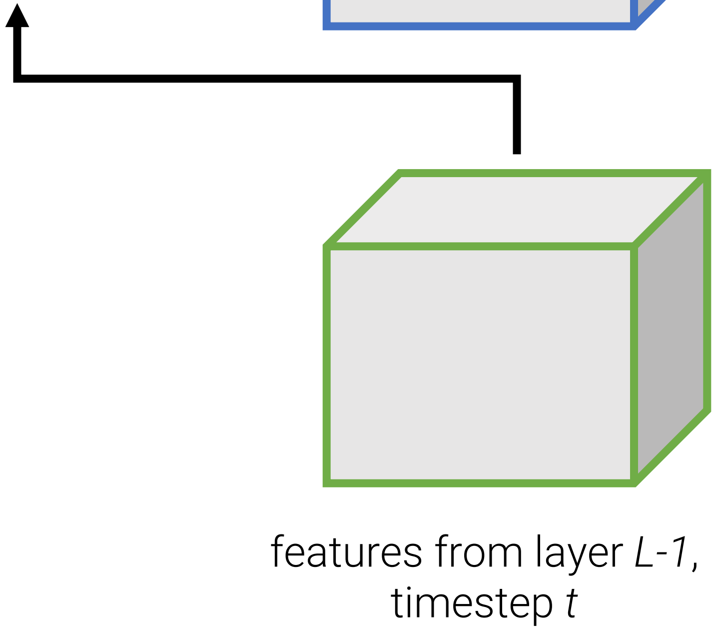
Higher Order Recurrent Convolutional Network



Higher order recurrence

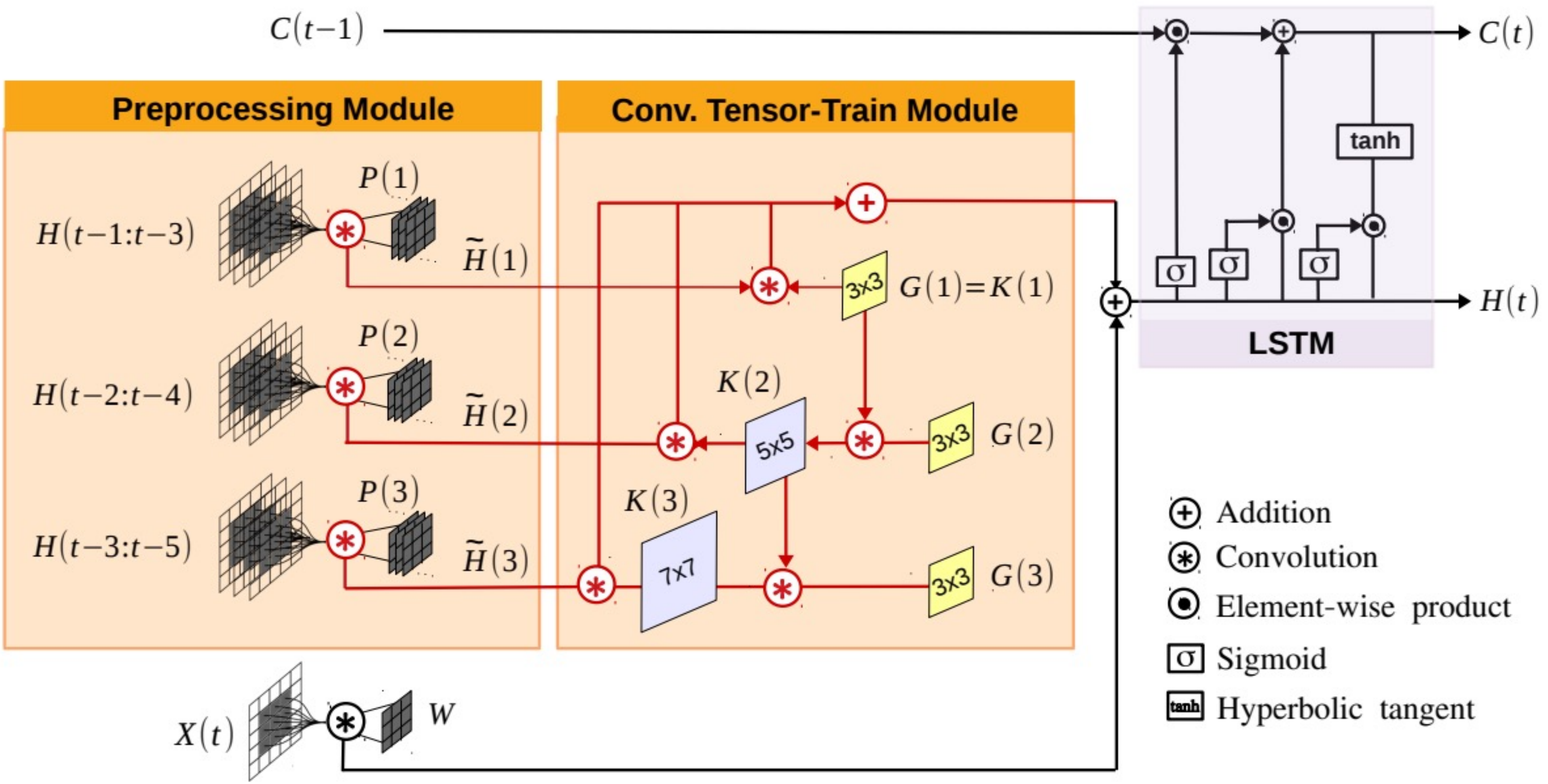
$$\mathbf{H}_t = f_{\mathbf{w}}(\mathbf{X}_t; \phi(\mathbf{H}_{t-1}, \ldots, \mathbf{H}_{t-S}))$$

How to aggregate paste states ? => ϕ ?



Desiderata for $\boldsymbol{\phi}$:
- spatial structure is preserved
- receptive field increases (more context) with earlier states
- complexities (in space and time) grow at most linearly with *S*

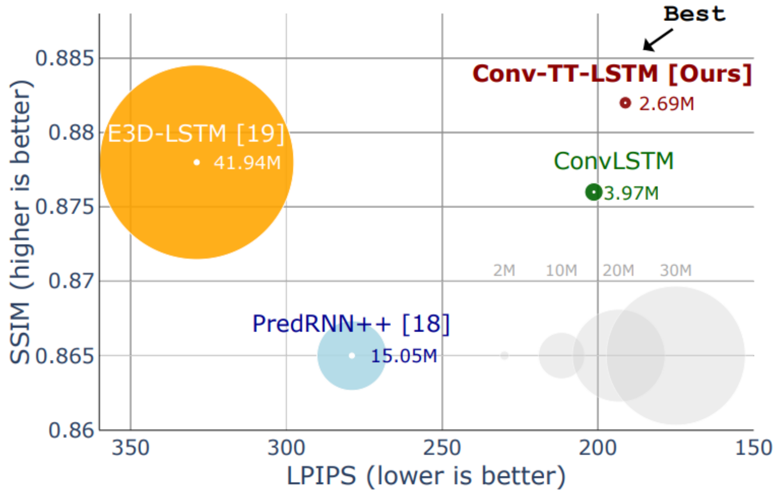
Convolutional Tensor-Train LSTM



Su et.al, "Convolutional Tensor-Train LSTM for Spatio-Temporal Learning", NeurIPS 2020

Convolutional Tensor-Train LSTM

Model	Input Ratio	
	Front 25%	Front 50%
3D-CNN*	9.11	10.30
E3D-LSTM* [7]	14.59	22.73
3D-CNN	13.26	20.72
ConvLSTM	15.46	21.97
Conv-TT-LSTM (ours)	**19.53**	**30.05**

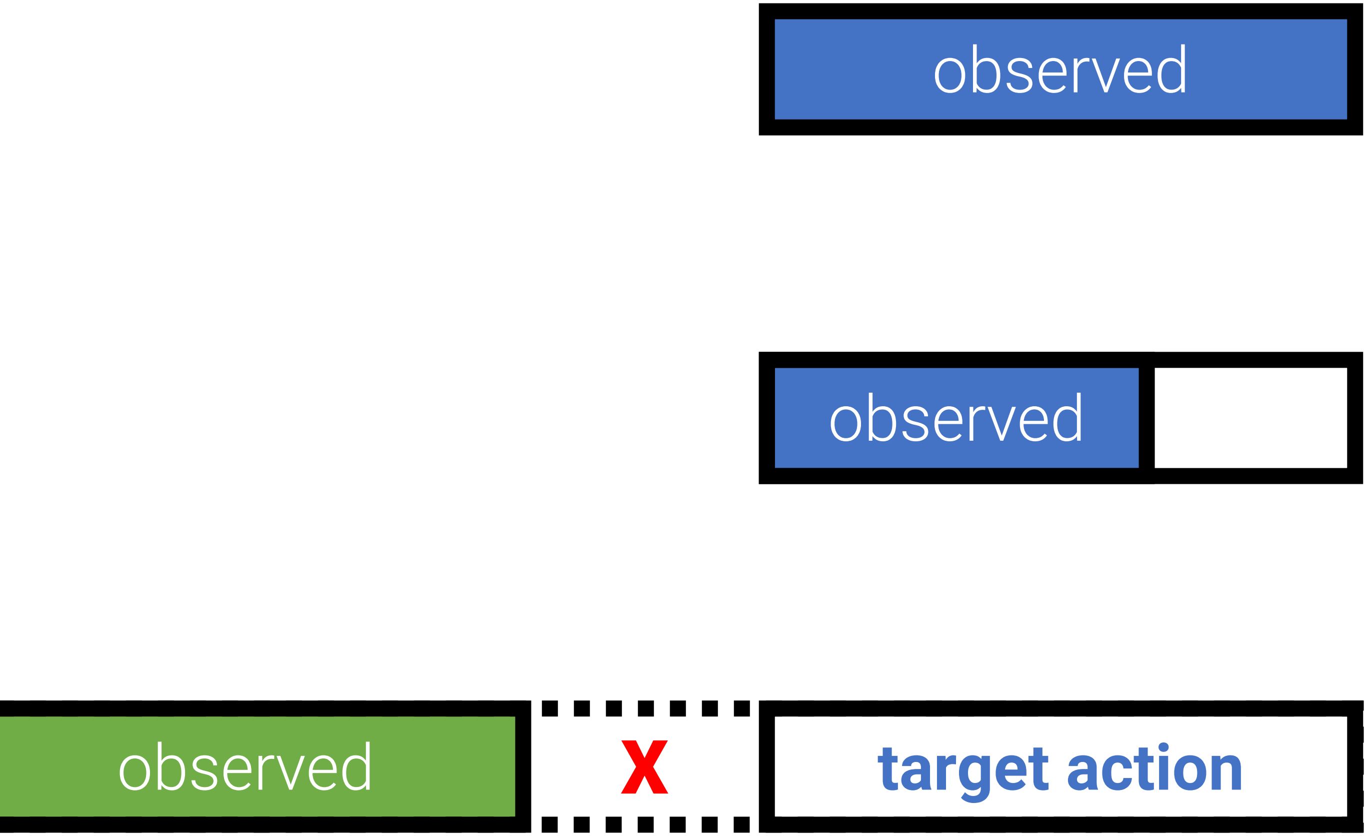


Multi-Frame Video Prediction on KTH action dataset: better performance while having a fraction of parameters

Model	Input	Dropping	Holding	MovingLR	MovingRL	Picking	Poking	Pouring	Putting	Showing	Tearing
3D-CNN		8.5	4.7	25.8	32.6	7.5	2.9	1.9	10.3	14.0	14.5
ConvLSTM	25%	8.5	**7.0**	27.4	38.8	16.8	5.9	1.9	12.0	7.0	21.2
Conv-TT-LSTM		**11.5**	4.7	**33.9**	**40.8**	**16.8**	**5.9**	**5.7**	**13.6**	**20.9**	**26.0**
3D-CNN		14.6	11.6	45.2	57.1	16.8	8.8	11.3	17.4	16.3	26.0
ConvLSTM	50%	21.5	7.0	43.5	47.0	15.9	**14.7**	5.7	20.7	16.3	30.8
Conv-TT-LSTM		**24.6**	**11.6**	**56.5**	**57.1**	**27.6**	5.9	**13.2**	**25.5**	**37.2**	**46.2**

Table 1: **Per-activity accuracy of early activity recognition on the Something-Something V2 dataset.** We used 41 categories for training. For per-activity evaluation, the 41 categories are grouped into 10 similar activities. The activity mapping are described in [21]. Our model substantially outperforms 3D-CNN and ConvLSTM on long-term dynamics such as Moving or Tearing, while achieves marginal improvement on static activities such as Holding or Pouring.

Early Action Recognition vs Action Anticipation



Action Recognition (= Trimmed Video Classification with Action Labels)

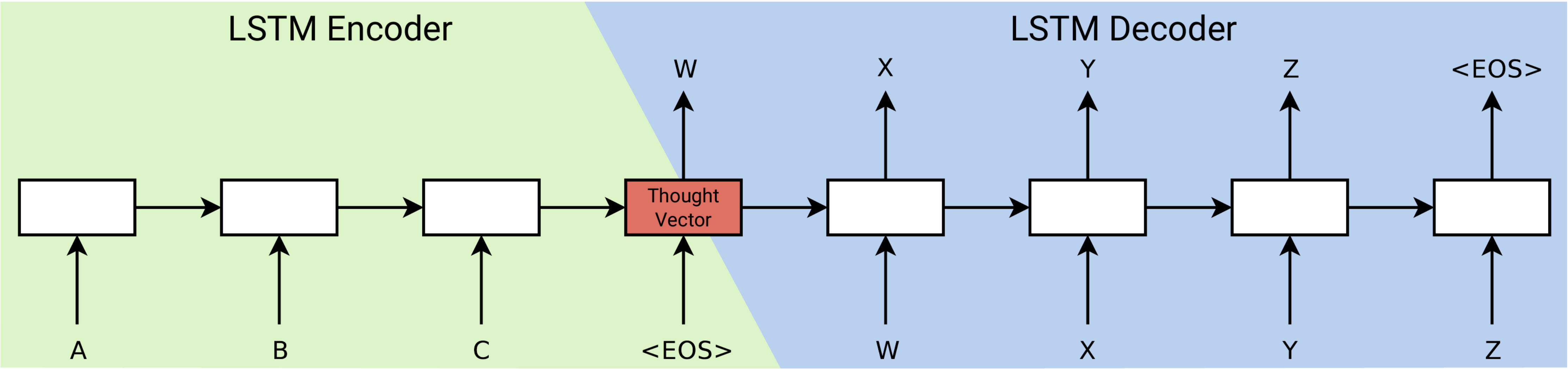
Early Action Recognition

Action Anticipation/Prediction

only pre-action that
may lead to target is observed

Is this a classification task ?

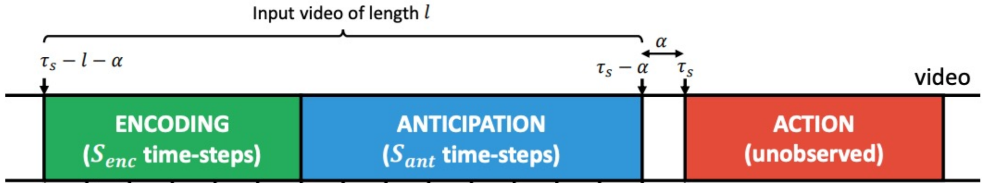
Sequence to Sequence Learning

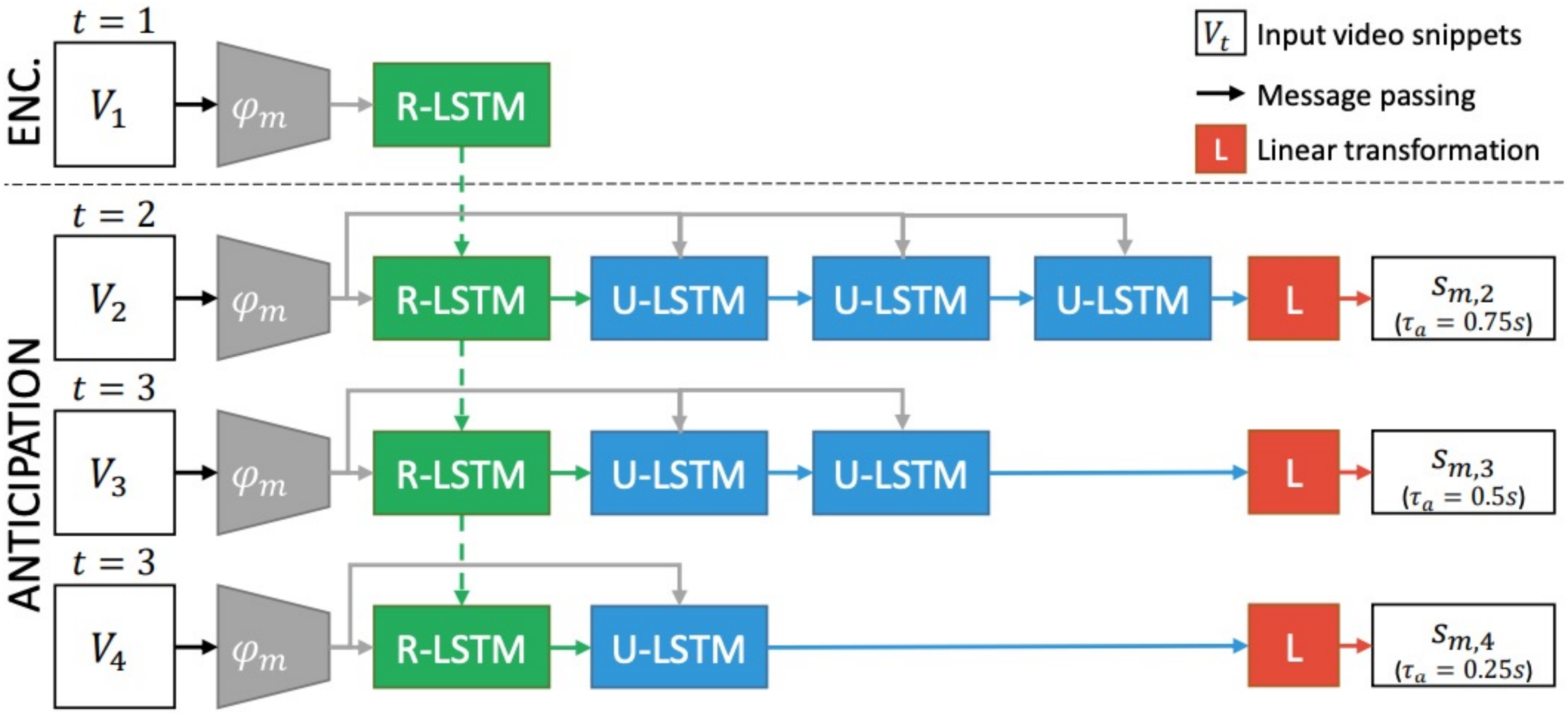


- **Two 4-Layer LSTMs** for encoding/decoding the source/target sentence
- Encoding operates in **reverse order** to introduce short-term dependencies
- Intermediate representation produced by the encoder is called **thought vector**
- Encoding using 1000 dim. word embeddings, decoding via **beam search**
- First end-to-end system that outperforms rule-based models $\Rightarrow$ deployment

Sutskever, Vinyals and Le: Sequence to Sequence Learning with Neural Networks. NIPS, 2014.

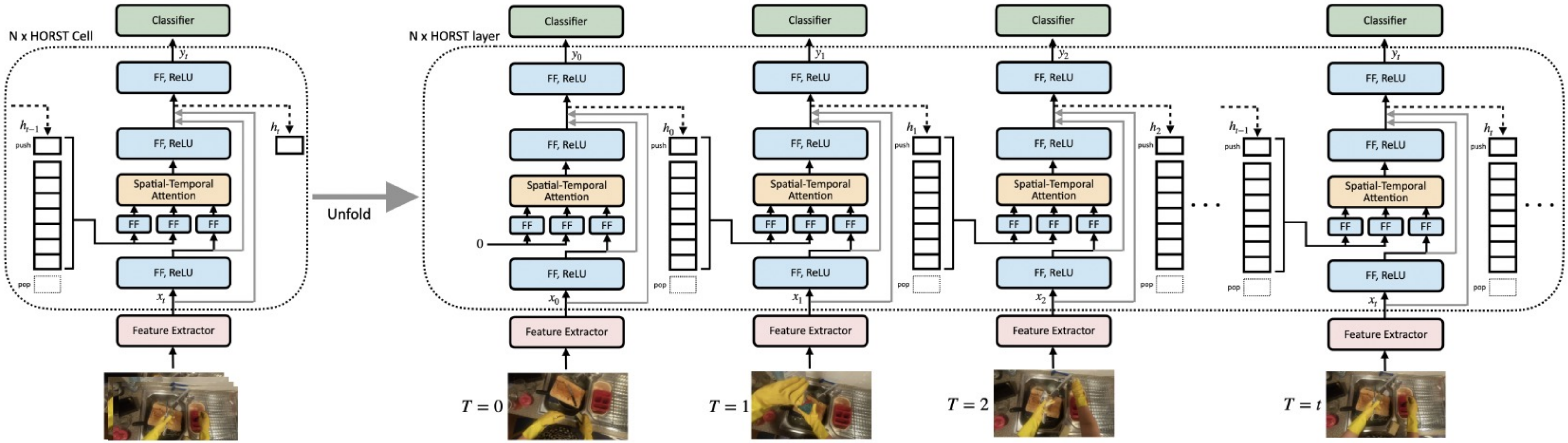
Rolling-Unrolling LSTM





Furnari and Farinella, "What Would You Expect? Anticipating Egocentric Actions With Rolling-Unrolling LSTMs and Modality Attention", ICCV 2019

Higher Order Recurrent Space-Time Transformer

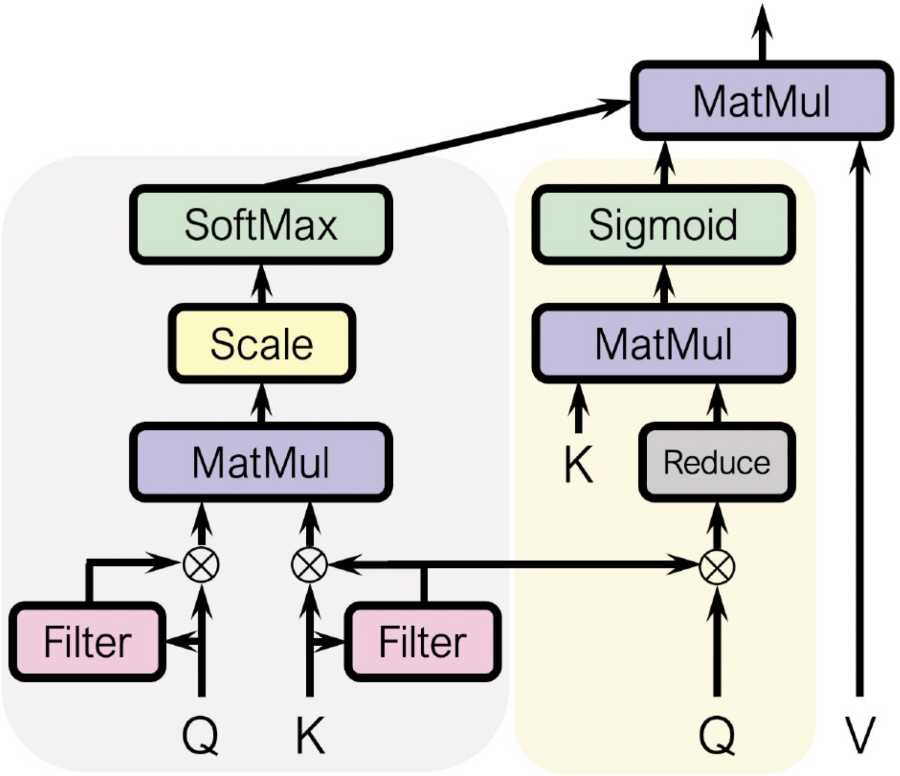


FF is Conv2D-LayerNorm

- *S*-order model: maintains a fifo queue of *S* past states
- Aggregation function ϕ is a spatial-temporal factorized self-attention (full space-time is $(S \cdot H \cdot W)^2$ ops !!)

Tai and Lanz et.al, "Higher Order Recurrent Network with Space-Time Attention for Video Early Action Recognition", ICIP 2022

Higher Order Recurrent Space-Time Transformer



spatial branch temporal

Spatial-Temporal factorized attention:

$$\mathrm{STATT}(\mathbf{Q},\mathbf{K},\mathbf{V}) = (\mathcal{S}(\mathbf{Q},\mathbf{K}) \otimes \mathcal{T}(\mathbf{Q},\mathbf{K})) \cdot \mathbf{V}$$

$$\mathcal{T}(\mathbf{Q},\mathbf{K}) = \mathrm{softmax}(\frac{(f_Q(\mathbf{Q}) \cdot \mathbf{Q})^\top (f_K(\mathbf{K}) \cdot \mathbf{K})}{\sqrt{C}})$$

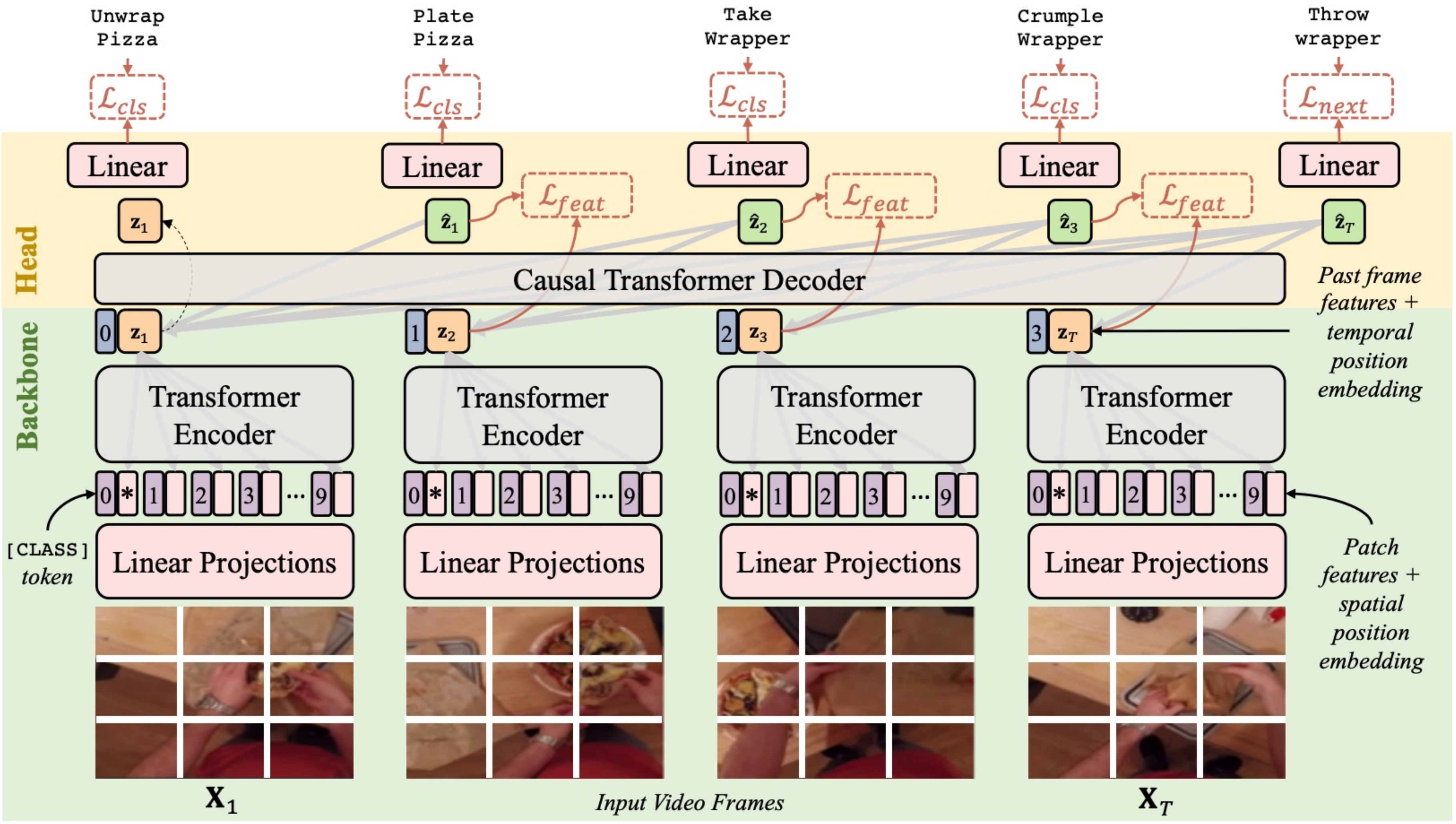
$$\mathcal{S}(\mathbf{Q},\mathbf{K}) = \mathrm{sigmoid}(\frac{(\mathrm{AvgPool}(f_K(\mathbf{K}) \cdot \mathbf{Q})^\top \mathbf{K})}{\sqrt{C}})$$

where $f_Q(\mathbf{Q}) = \mathrm{sigmoid}(\mathbf{w}_Q * [\mathbf{Q}_{\max}, \mathbf{Q}_{\mathrm{avg}}])$

$$\mathrm{ATT}(\mathbf{Q},\mathbf{K},\mathbf{V}) = \mathrm{softmax}(\frac{\mathbf{Q}^\top \mathbf{K}}{\sqrt{C}}) \cdot \mathbf{V}$$

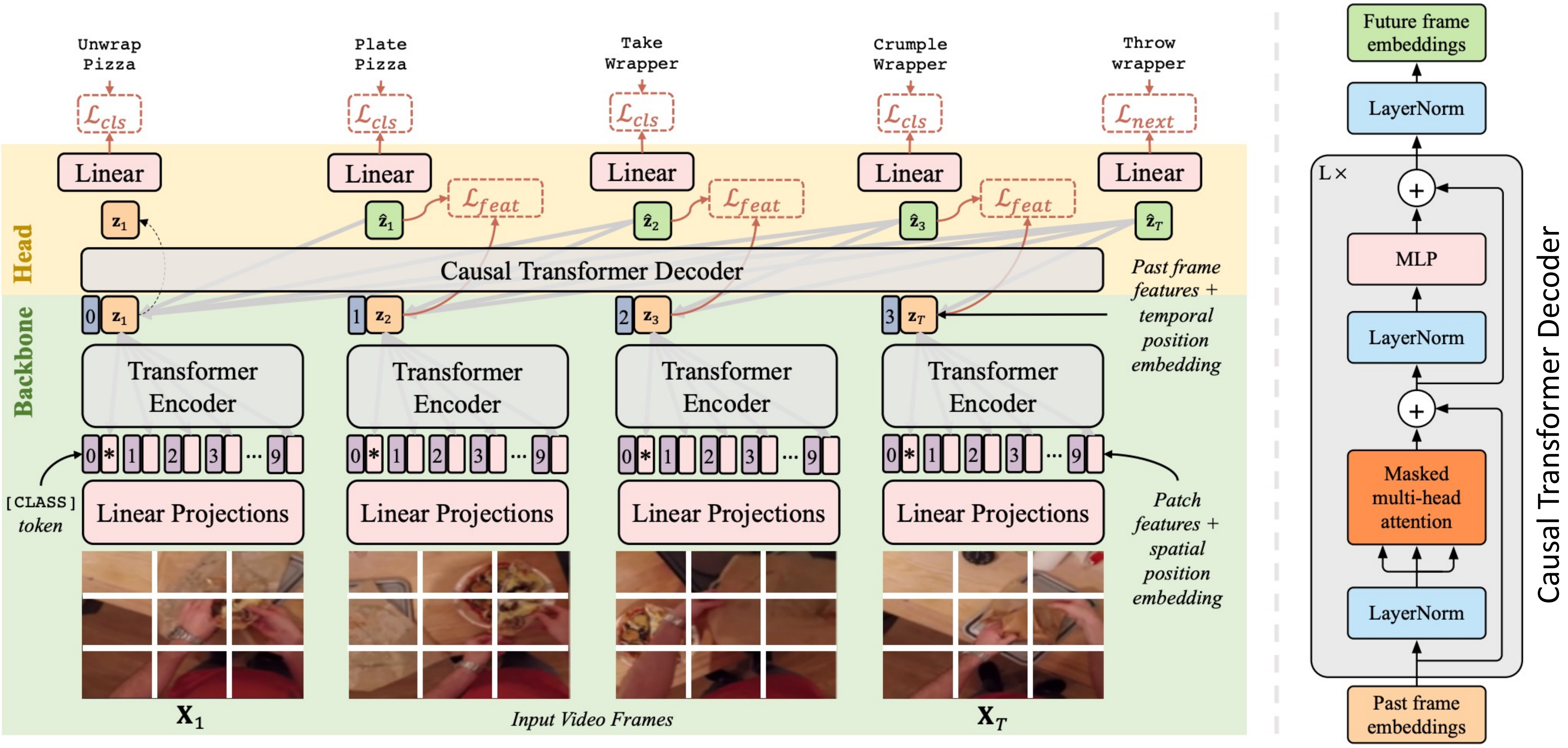
Tai and Lanz et.al, “Higher Order Recurrent Network with Space-Time Attention for Video Early Action Recognition”, ICIP 2022

Anticipative Video Transformer



Girdhar and Grauman, "Anticipative Video Transformer", ICCV 2021

Anticipative Video Transformer



Girdhar and Grauman, “Anticipative Video Transformer”, ICCV 2021

Anticipative Video Transformer

Head	Backbone	Init	Top-1	Top-5	Recall
RULSTM [24]	TSN	IN1k	13.1	30.8	12.5
ActionBanks [77]	TSN	IN1k	12.3	28.5	13.1
AVT-h	TSN	IN1k	13.1	28.1	13.5
AVT-h	AVT-b	IN21+1k	12.5	30.1	**13.6**
AVT-h	irCSN152	IG65M	**14.4**	**31.7**	13.2

Table 4: EK55 using only RGB modality for action anticipation. AVT performs comparably, and outperforms when combined with a backbone pretrained on large weakly labeled dataset.

		Overall			Unseen Kitchen			Tail Classes		
Split	Method	Verb	Noun	Act	Verb	Noun	Act	Verb	Noun	Act
Val	chance	6.4	2.0	0.2	14.4	2.9	0.5	1.6	0.2	0.1
	RULSTM [14]	27.8	30.8	14.0	28.8	**27.2**	**14.2**	19.8	22.0	11.1
	AVT+ (TSN)	25.5	31.8	14.8	25.5	23.6	11.5	18.5	25.8	12.6
	AVT+	**28.2**	**32.0**	**15.9**	**29.5**	23.9	11.9	**21.1**	**25.8**	**14.1**
Test	chance	6.2	2.3	0.1	8.1	3.3	0.3	1.9	0.7	0.0
	RULSTM [14]	25.3	26.7	11.2	19.4	26.9	9.7	17.6	16.0	7.9
	TBN [100]	21.5	26.8	11.0	20.8	**28.3**	**12.2**	13.2	15.4	7.2
	AVT+	**25.6**	**28.8**	**12.6**	**20.9**	22.3	8.8	**19.0**	**22.0**	**10.1**
Challenge	IIE_MRG	25.3	26.7	11.2	19.4	26.9	9.7	17.6	16.0	7.9
	NUS_CVML [76]	21.8	30.6	12.6	17.9	27.0	10.5	13.6	20.6	8.9
	ICL+SJTU [35]	**36.2**	32.2	13.4	**27.6**	24.2	10.1	**32.1**	**29.9**	11.9
	Panasonic [98]	30.4	**33.5**	14.8	21.1	27.1	10.2	24.6	27.5	12.7
	AVT++	25.2	32.0	**16.5**	20.4	**27.9**	**12.8**	17.6	23.5	**13.6**

Table 3: EK100 val and test sets using all modalities. We split the test comparisons between published work and CVPR'21 challenge submissions. We outperform prior work including all challenge submissions, with especially significant gains on tail classes. Performance is reported using class-mean recall@5. AVT+ and AVT++ late fuse predictions from multiple modalities; please see text for details.

Girdhar and Grauman, "Anticipative Video Transformer", ICCV 2021

EPIC-KITCHENS-100 Action Anticipation

Organized by antonino - Current server time: Sept. 3, 2023, 5:53 p.m. UTC

▶ Current	End
2023 Open Testing Phase	Competition Ends
June 27, 2023, 8 a.m. UTC	Nov. 25, 2023, 11 p.m. UTC

Test Set (Mean Top-5 Recall)

#	User	Entries	Date of Last Entry	Team Name	SLS PT	SLS TL	SLS TD	Overall (%) Verb	Overall (%) Noun	Overall (%) Action	Unseen (%) Verb	Unseen (%) Noun	Unseen (%) Action	Tail (%) Verb	Tail (%) Noun	Tail (%) Action
1	latent	29	10/18/22	InAViT IHPC-AISG-LAHA	1.0 (2)	3.0 (2)	3.0 (2)	49.14 (1)	49.97 (1)	23.75 (1)	44.36 (1)	49.28 (1)	23.49 (1)	43.17 (1)	39.91 (1)	18.11 (1)
2	hrgdscs	7	06/01/22		2.0 (1)	3.0 (2)	3.0 (2)	37.91 (4)	41.71 (2)	20.43 (2)	27.94 (4)	37.07 (2)	18.27 (2)	32.43 (4)	36.09 (2)	17.11 (2)
3	corcovadoming	28	06/01/22	NVIDIA-UNIBZ	1.0 (2)	3.0 (2)	4.0 (1)	29.67 (10)	38.46 (4)	19.61 (3)	23.47 (8)	35.25 (4)	16.41 (3)	23.48 (10)	31.11 (6)	16.63 (4)
4	shawn0822	22	06/01/22	ICL-SJTU	2.0 (1)	4.0 (1)	4.0 (1)	41.96 (3)	35.74 (5)	19.53 (4)	33.35 (3)	26.80 (13)	15.85 (5)	41.01 (3)	33.22 (4)	16.87 (3)
5	PCO-PSNRD	7	05/30/22	PCO-PSNRD	2.0 (1)	4.0 (1)	3.0 (2)	30.85 (6)	41.32 (3)	18.68 (5)	25.65 (6)	35.39 (3)	16.32 (4)	24.99 (6)	35.40 (3)	16.14 (5)
6	allenxuuu	1	12/20/21	2021 Open Testing Phase	2.0 (1)	4.0 (1)	4.0 (1)	29.88 (9)	30.40 (15)	17.35 (6)	25.08 (7)	26.08 (14)	14.14 (6)	24.60 (7)	23.68 (12)	14.30 (7)
7	Shawn0822-ICL-SJTU	1	12/20/21	2021 Open Testing Phase	1.0 (2)	4.0 (1)	3.0 (2)	42.32 (2)	34.60 (6)	17.02 (7)	33.36 (2)	25.94 (16)	12.84 (8)	42.47 (2)	31.37 (5)	15.56 (6)
8	shef-AVT-FB-UT	1	12/20/21	2021 Open Testing Phase	2.0 (1)	4.0 (1)	4.0 (1)	26.69 (13)	32.33 (10)	16.74 (8)	21.03 (12)	27.64 (7)	12.89 (7)	19.28 (13)	24.03 (10)	13.81 (8)
9	richard61	8	05/31/22		2.0 (1)	4.0 (1)	4.0 (1)	27.60 (11)	32.45 (9)	16.68 (9)	20.10 (14)	28.13 (5)	12.42 (11)	20.12 (12)	23.89 (11)	13.80 (10)
10	Zeyun-Zhong	12	06/01/22	KIT-IAR-IOSB	1.0 (2)	4.0 (1)	3.0 (2)	30.03 (8)	33.45 (8)	16.65 (10)	23.16 (9)	27.20 (8)	12.63 (10)	23.65 (9)	26.86 (9)	13.80 (9)
11	AVT-FB-UT	1	12/15/21	CVPR 2021 Challenges	2.0 (1)	4.0 (1)	4.0 (1)	25.25 (16)	32.04 (12)	16.53 (11)	20.41 (13)	27.90 (6)	12.79 (9)	17.63 (15)	23.47 (13)	13.62 (11)
12	zhh6	9	11/08/22		2.0 (1)	4.0 (1)	4.0 (1)	27.43 (12)	31.53 (13)	15.87 (12)	22.28 (10)	26.90 (11)	11.70 (12)	20.21 (11)	23.14 (14)	12.92 (12)
13	Panasonic-CNSIC-PSNRD	1	12/15/21	CVPR 2021 Challenges	1.0 (2)	4.0 (1)	3.0 (2)	30.38 (7)	33.50 (7)	14.82 (13)	21.08 (11)	27.11 (9)	10.21 (15)	24.57 (8)	27.45 (8)	12.69 (13)
14	ICL-SJTU	1	12/15/21	CVPR 2021 Challenges	1.0 (2)	4.0 (1)	3.0 (2)	36.15 (5)	32.20 (11)	13.39 (14)	27.60 (5)	24.24 (17)	10.05 (16)	32.06 (5)	29.87 (7)	11.88 (14)
15	NUS-CVML	1	12/15/21	CVPR 2021 Challenges	1.0 (2)	4.0 (1)	3.0 (2)	21.76 (17)	30.59 (14)	12.55 (15)	17.86 (17)	27.04 (10)	10.46 (13)	13.59 (17)	20.62 (15)	8.85 (15)
16	qzhb	19	11/12/22		1.0 (2)	4.0 (1)	3.0 (2)	25.67 (14)	26.49 (17)	11.64 (16)	19.31 (16)	26.05 (15)	10.25 (14)	18.05 (14)	15.71 (17)	8.42 (16)
17	RULSTM-FUSION	1	12/15/21	CVPR 2021 Challenges	1.0 (2)	4.0 (1)	3.0 (2)	25.25 (15)	26.69 (16)	11.19 (17)	19.36 (15)	26.87 (12)	9.65 (17)	17.56 (16)	15.97 (16)	7.92 (17)
18	EPIC-CHANCE-BASELINE	1	12/15/21	CVPR 2021 Challenges	0.0 (3)	1.0 (3)	3.0 (2)	6.17 (18)	2.28 (18)	0.14 (18)	8.14 (18)	3.28 (18)	0.31 (18)	1.87 (18)	0.66 (18)	0.03 (18)

Table 1. Individual model performance on validation set, measured in mean top-5 action recall (MT5R) at 1s, of various modalities using different modelings and backbones.

Model	Modality	Backbone	MT5R (%)
HORST	RGB	Swin-B	18.42
HORST	RGB	ConvNeXt	17.09
MPNNEL	RGB	Swin-B	17.05
MPNNEL (CTP)	RGB	Swin-B	18.18
MPNNEL (TB)	RGB	Swin-B	17.05
MPNNEL	RGB	ConvNeXt	17.18
MPNNEL (CTP)	RGB	ConvNeXt	18.54
MPNNEL (TB)	RGB	ConvNeXt	18.09
HORST	Flow	Swin-B	7.95
HORST	Flow	ConvNeXt	7.36
HORST	Flow (Snippets)	Swin-B	6.61
HORST	Flow (Snippets)	ConvNeXt	8.06
MPNNEL	Flow	Swin-B	-
MPNNEL (CTP)	Flow	Swin-B	6.66
MPNNEL (TB)	Flow	Swin-B	-
MPNNEL	Flow	ConvNeXt	7.59
MPNNEL (CTP)	Flow	ConvNeXt	8.74
MPNNEL (TB)	Flow	ConvNeXt	8.18
HORST	Obj	None	8.72
MPNNEL	Obj	None	9.69
MPNNEL (CTP)	Obj	None	8.80
MPNNEL (TB)	Obj	None	8.99
HORST	Masked-RGB	Swin-B	12.03
HORST	Masked-RGB	ConvNeXt	11.30
MPNNEL	Masked-RGB	Swin-B	9.22
MPNNEL (CTP)	Masked-RGB	Swin-B	7.87
MPNNEL (TB)	Masked-RGB	Swin-B	9.57
MPNNEL	Masked-RGB	ConvNeXt	9.65
MPNNEL (CTP)	Masked-RGB	ConvNeXt	8.53
MPNNEL (TB)	Masked-RGB	ConvNeXt	10.30

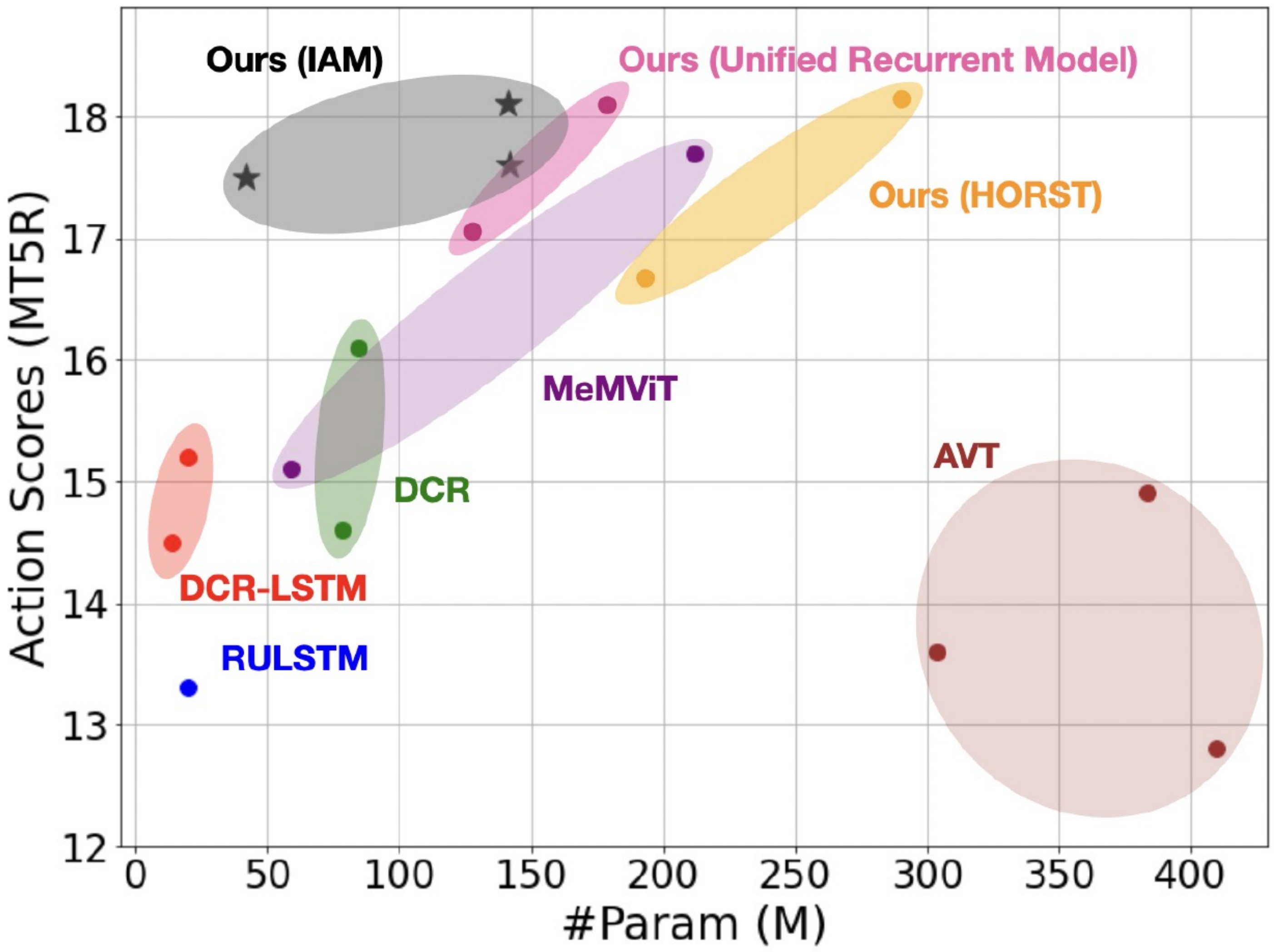
Table 2. Test accuracy of model ensemble.

Model	MT5R (%)
(a) HORST Family with all modalities	17.47
(b) MPNNEL Family with all modalities	18.19
(a) + (b)	19.52
(a) + (b) and weightings 1.2x on all RGB models	**19.61**

Test Set (Mean Top-5 Recall)

#	User	Entries	Date of Last Entry	Team Name	SLS			Overall (%)			Unseen (%)			Tail (%)		
					PT ▲	TL ▲	TD ▲	Verb ▲	Noun ▲	Action ▲	Verb ▲	Noun ▲	Action ▲	Verb ▲	Noun ▲	Action ▲
1	latent	29	10/18/22	InAViT IHPC-AISG-LAHA	1.0 (2)	3.0 (2)	3.0 (2)	49.14 (1)	49.97 (1)	23.75 (1)	44.36 (1)	49.28 (1)	23.49 (1)	43.17 (1)	39.91 (1)	18.11 (1)
2	hrgdscs	7	06/01/22		2.0 (1)	3.0 (2)	3.0 (2)	37.91 (4)	41.71 (2)	20.43 (2)	27.94 (4)	37.07 (2)	18.27 (2)	32.43 (4)	36.09 (2)	17.11 (2)
3	corcovadoming	28	06/01/22	NVIDIA-UNIBZ	1.0 (2)	3.0 (2)	4.0 (1)	29.67 (10)	38.46 (4)	19.61 (3)	23.47 (8)	35.25 (4)	16.41 (3)	23.48 (10)	31.11 (6)	16.63 (4)
4	shawn0822	22	06/01/22	ICL-SJTU	2.0 (1)	4.0 (1)	4.0 (1)	41.96 (3)	35.74 (5)	19.53 (4)	33.35 (3)	26.80 (13)	15.85 (5)	41.01 (3)	33.22 (4)	16.87 (3)
5	PCO-PSNRD	7	05/30/22	PCO-PSNRD	2.0 (1)	4.0 (1)	3.0 (2)	30.85 (6)	41.32 (3)	18.68 (5)	25.65 (6)	35.39 (3)	16.32 (4)	24.99 (6)	35.40 (3)	16.14 (5)
6	allenxuuu	1	12/20/21	2021 Open Testing Phase	2.0 (1)	4.0 (1)	4.0 (1)	29.88 (9)	30.40 (15)	17.35 (6)	25.08 (7)	26.08 (14)	14.14 (6)	24.60 (7)	23.68 (12)	14.30 (7)
7	Shawn0822-ICL-SJTU	1	12/20/21	2021 Open Testing Phase	1.0 (2)	4.0 (1)	3.0 (2)	42.32 (2)	34.60 (6)	17.02 (7)	33.36 (2)	25.94 (16)	12.84 (8)	42.47 (2)	31.37 (5)	15.56 (6)
8	shef-AVT-FB-UT	1	12/20/21	2021 Open Testing Phase	2.0 (1)	4.0 (1)	4.0 (1)	26.69 (13)	32.33 (10)	16.74 (8)	21.03 (12)	27.64 (7)	12.89 (7)	19.28 (13)	24.03 (10)	13.81 (8)
9	richard61	8	05/31/22		2.0 (1)	4.0 (1)	4.0 (1)	27.60 (11)	32.45 (9)	16.68 (9)	20.10 (14)	28.13 (5)	12.42 (11)	20.12 (12)	23.89 (11)	13.80 (10)
10	Zeyun-Zhong	12	06/01/22	KIT-IAR-IOSB	1.0 (2)	4.0 (1)	3.0 (2)	30.03 (8)	33.45 (8)	16.65 (10)	23.16 (9)	27.20 (8)	12.63 (10)	23.65 (9)	26.86 (9)	13.80 (9)
11	AVT-FB-UT	1	12/15/21	CVPR 2021 Challenges	2.0 (1)	4.0 (1)	4.0 (1)	25.25 (16)	32.04 (12)	16.53 (11)	20.41 (13)	27.90 (6)	12.79 (9)	17.63 (15)	23.47 (13)	13.62 (11)
12	zhh6	9	11/08/22		2.0 (1)	4.0 (1)	4.0 (1)	27.43 (12)	31.53 (13)	15.87 (12)	22.28 (10)	26.90 (11)	11.70 (12)	20.21 (11)	23.14 (14)	12.92 (12)
13	Panasonic-CNSIC-PSNRD	1	12/15/21	CVPR 2021 Challenges	1.0 (2)	4.0 (1)	3.0 (2)	30.38 (7)	33.50 (7)	14.82 (13)	21.08 (11)	27.11 (9)	10.21 (15)	24.57 (8)	27.45 (8)	12.69 (13)
14	ICL-SJTU	1	12/15/21	CVPR 2021 Challenges	1.0 (2)	4.0 (1)	3.0 (2)	36.15 (5)	32.20 (11)	13.39 (14)	27.60 (5)	24.24 (17)	10.05 (16)	32.06 (5)	29.87 (7)	11.88 (14)
15	NUS-CVML	1	12/15/21	CVPR 2021 Challenges	1.0 (2)	4.0 (1)	3.0 (2)	21.76 (17)	30.59 (14)	12.55 (15)	17.86 (17)	27.04 (10)	10.46 (13)	13.59 (17)	20.62 (15)	8.85 (15)
16	qzhb	19	11/12/22		1.0 (2)	4.0 (1)	3.0 (2)	25.67 (14)	26.49 (17)	11.64 (16)	19.31 (16)	26.05 (15)	10.25 (14)	18.05 (14)	15.71 (17)	8.42 (16)
17	RULSTM-FUSION	1	12/15/21	CVPR 2021 Challenges	1.0 (2)	4.0 (1)	3.0 (2)	25.25 (15)	26.69 (16)	11.19 (17)	19.36 (15)	26.87 (12)	9.65 (17)	17.56 (16)	15.97 (16)	7.92 (17)
18	EPIC-CHANCE-BASELINE	1	12/15/21	CVPR 2021 Challenges	0.0 (3)	1.0 (3)	3.0 (2)	6.17 (18)	2.28 (18)	0.14 (18)	8.14 (18)	3.28 (18)	0.31 (18)	1.87 (18)	0.66 (18)	0.03 (18)

EPIC-Kitchens-100 (validation set)



- "Inductive Attention for Video Action Anticipation", arXiv 2023
- "Unified recurrence modeling for video action anticipation", ICPR 2022
- "Higher Order Recurrent Network with Space-Time Attention for Video Early Action Recognition", ICIP 2022 - extension

From Action Recognition to Action Anticipation

Oswald Lanz

Univ. of Bolzano

https://vision.inf.unibz.it