Tsinghua University

Advanced Computer Vision

THU×SENSETIME – 80231202

sensetime

Chapter 2 - Section 9

Learning from Videos

Dr Yali Li

Friday, April 16, 2021

Acknowledge : Han Wang , Zhongdao Wang

商汤 sensetime

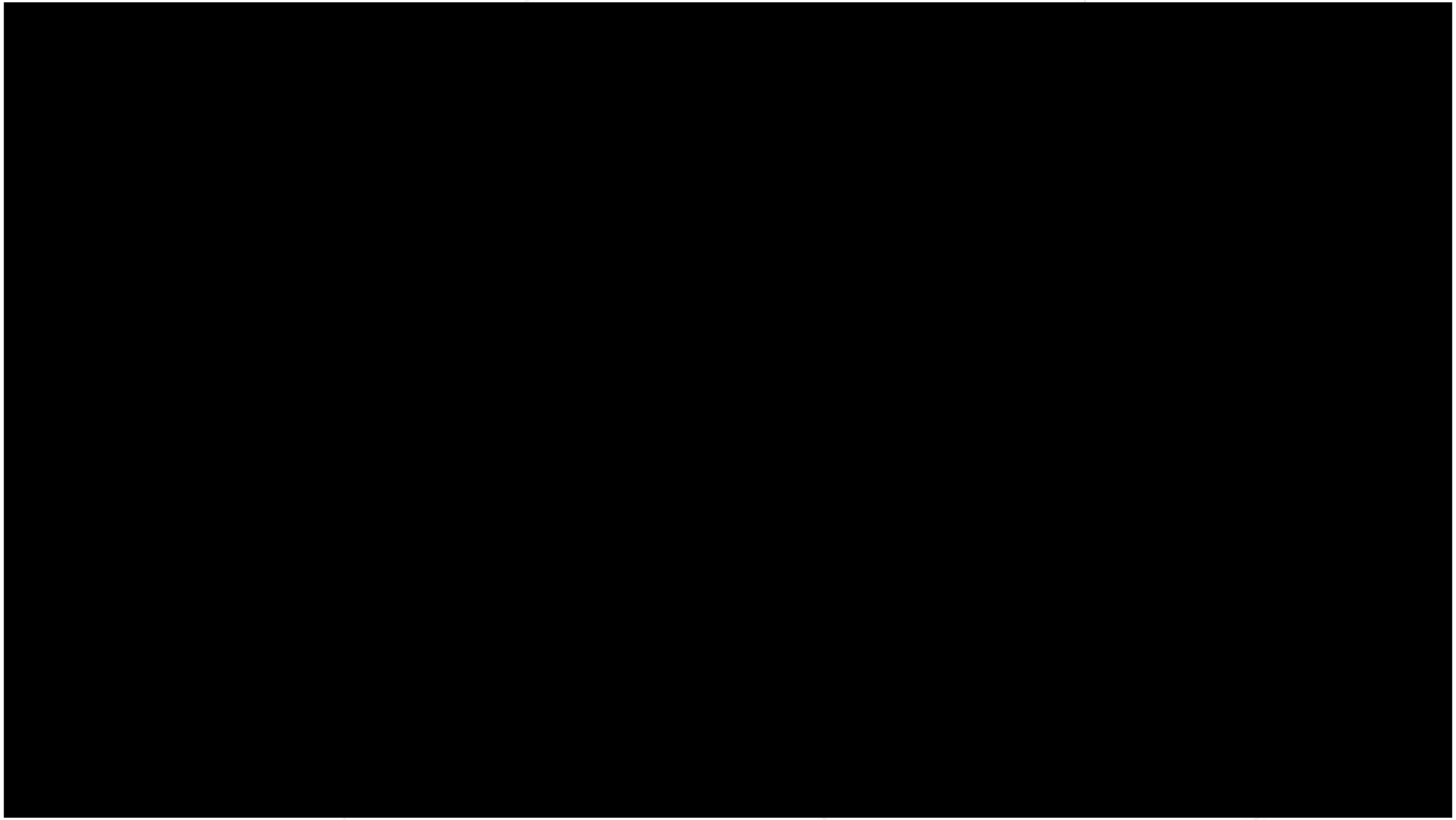
清華大學 Tsinghua University

Outline

- We collect videos everywhere
  - Websites, mobiles, etc.



Ref. [1] Author . Paper Title. Conference. / Git: www.github.com/XXX



https://b23.tv/OOymwr

Self-Driving Cars

https://b23.tv/3Rx3lr

Activity Understanding

https://cs.stanford.edu/people/karpathy/deepvideo/cnn_video_classify_demo.m4v

Video Classification

- Example tasks: video recognition vs image recognition

Images:

Recognize objects →

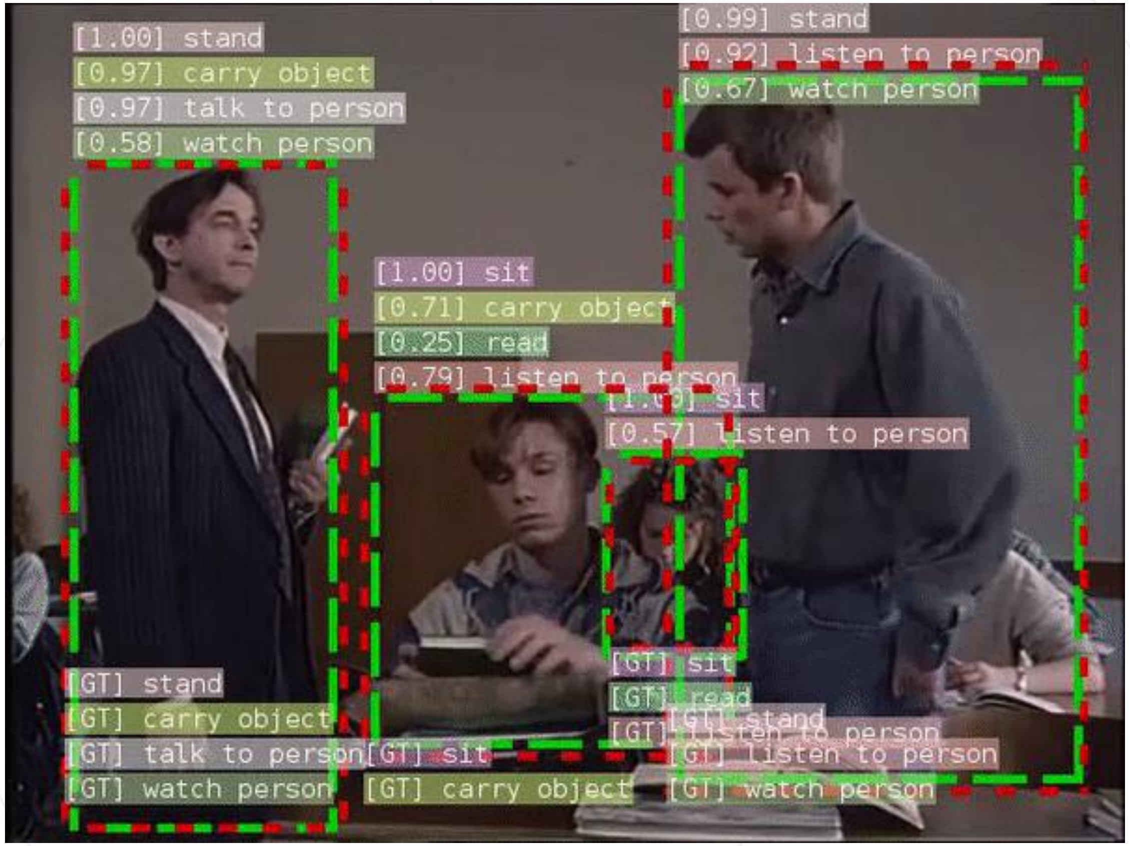
Dog
Cat
Horse
Cow

Videos:

Recognize actions →

Running
Jumping
Biking
Standing
Eating

- Example tasks: video classification /action recognition



Ref. [1] Author . Paper Title. Conference. / Git: www.github.com/XXX

Outline

Tsinghua University

sensetime

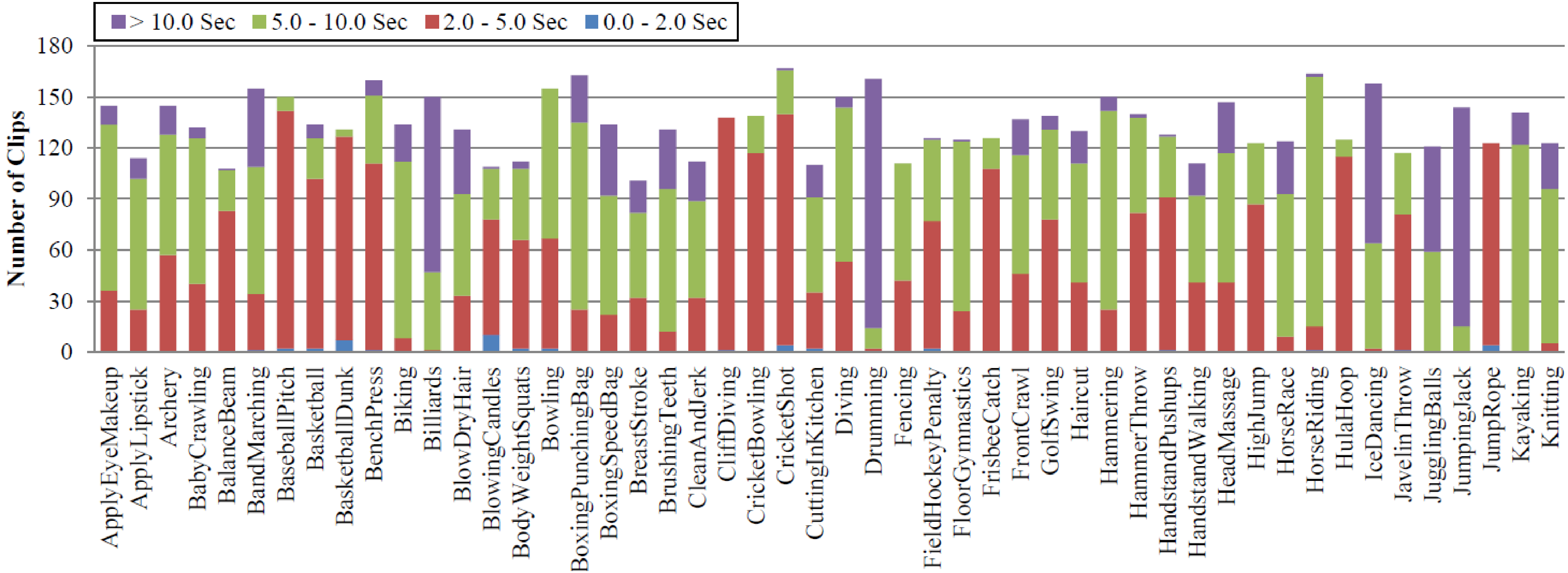
- UCF101
  - Youtube videos
  - 101 action classes, 13320 videos, ~27 hours of video data
  - Large variations in camera motion, object appearance and pose, viewpoint, background, illumination, etc.
  - HOG/HOF descriptors + SVM: yield an overall accuracy 43.9%。

K Soomro et al, *UCF101: A Dataset of 101 Human Action Classes From Videos in The Wild*, CRCV-TR-12-01, November, 2012.

https://www.crcv.ucf.edu/research/data-sets/ucf101/

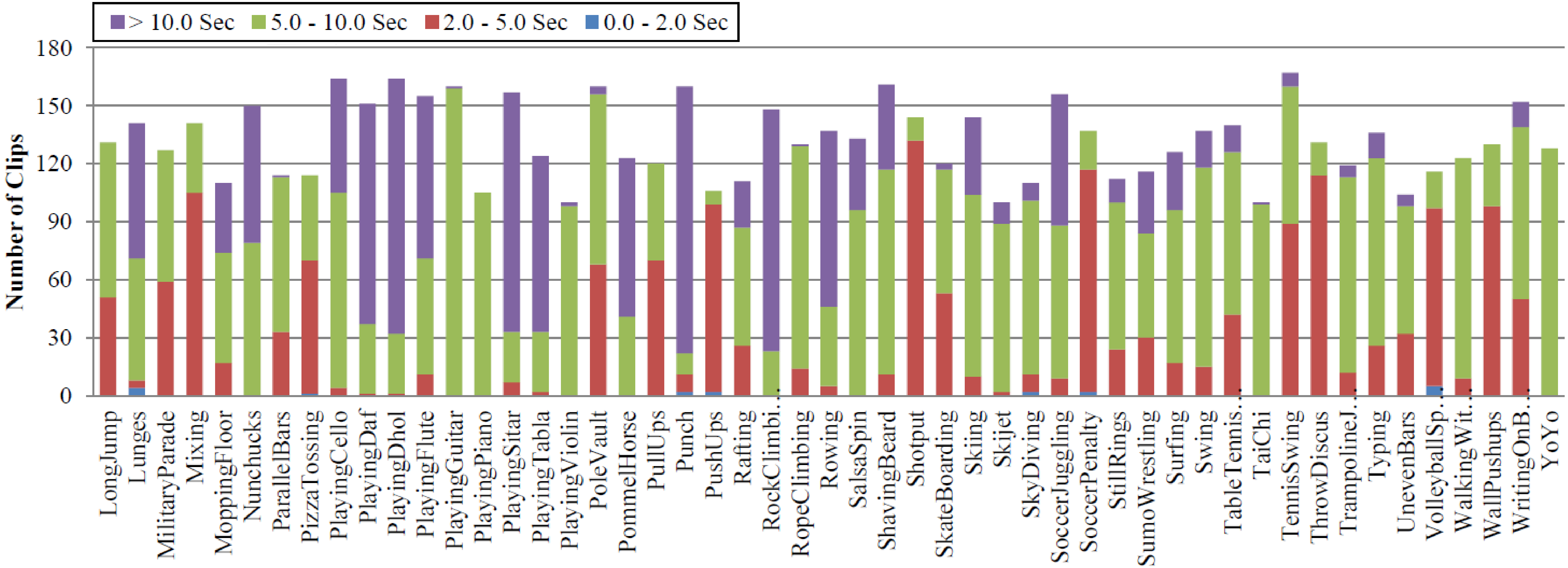
Video Classification - Datasets

- UCF101
  - 101 action classes, 13320 videos, ~27 hours of video data



https://www.crcv.ucf.edu/research/data-sets/ucf101/

- UCF101
  - 101 action classes, 13320 videos, ~27 hours of video data



https://www.crcv.ucf.edu/research/data-sets/ucf101/

Video Classification - Datasets

- Sports-1M
  - YouTube videos， 1,133,157 videos, 487 sports labels

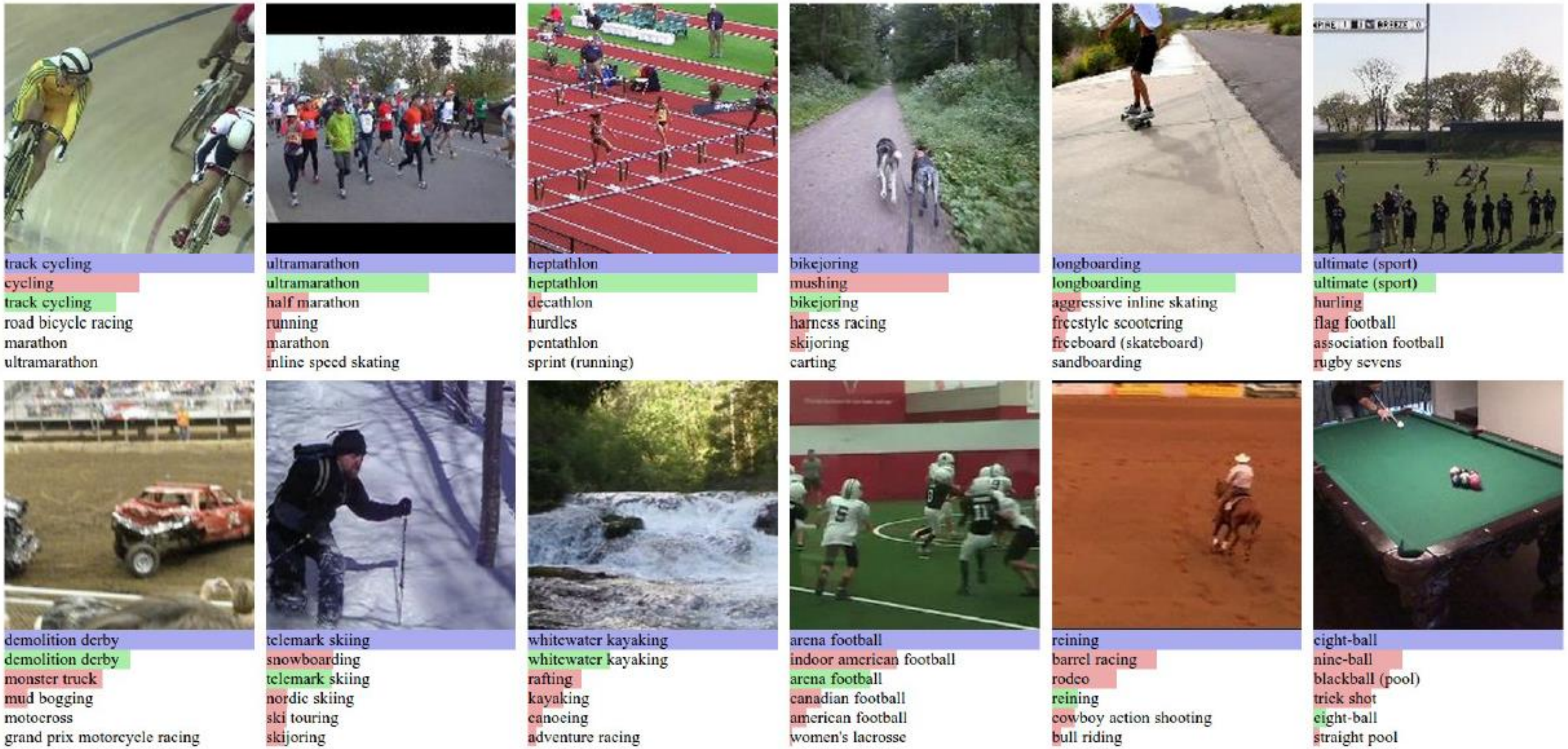
Video Classification - Datasets

- Sports-1M
  - 1 million YouTube videos, belonging to 487 classes.
  - 1000-3000 videos per class
  - ~5% of the videos are annotated with more than one class.

boomerang
boxing
bowling
candlepin bowling
bowls
skittles (sport)
ten-pin bowling
cycling

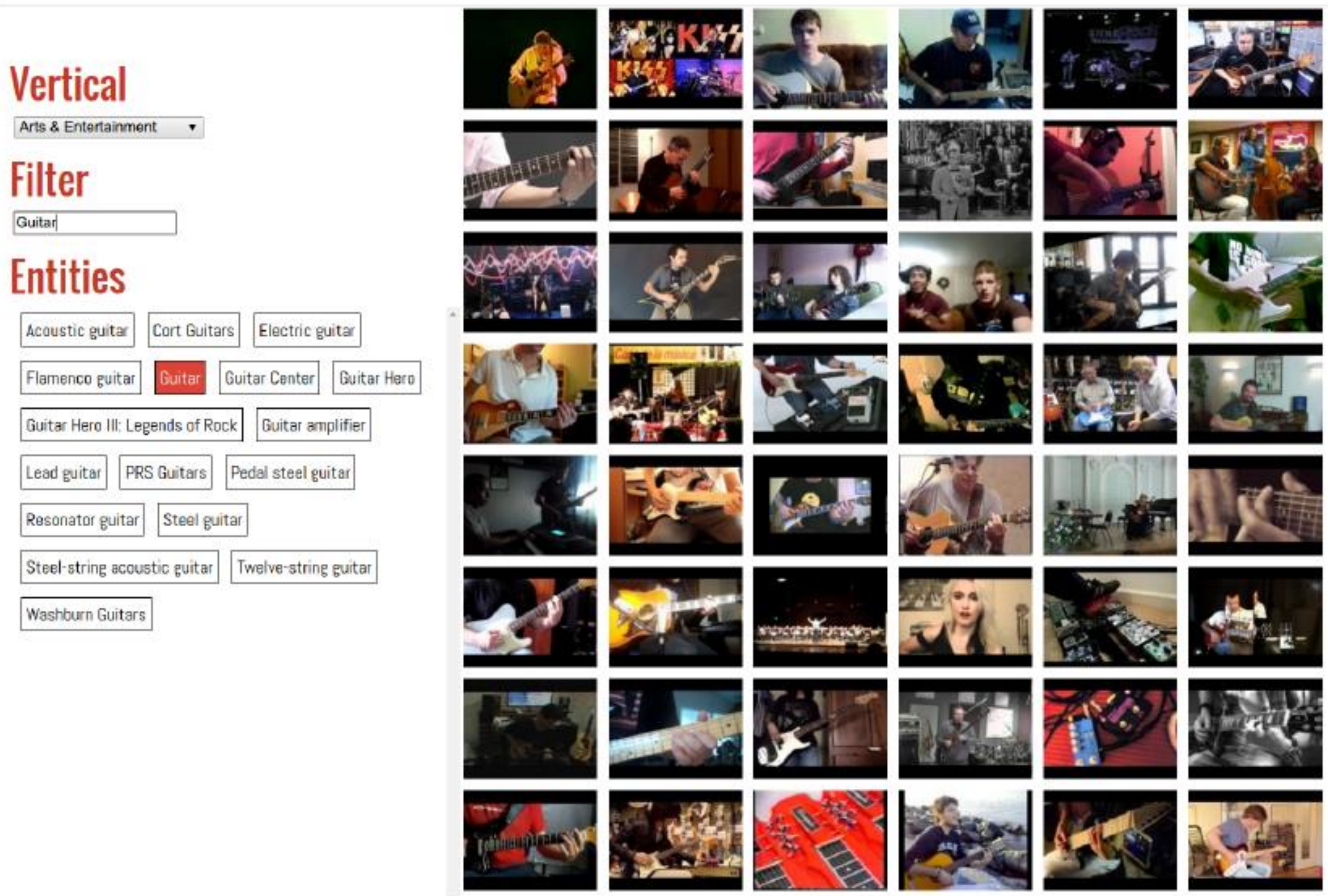
http://cs.stanford.edu/people/karpathy/deepvideo

- Sports-1M


track cycling
cycling
track cycling
road bicycle racing
marathon
ultramarathon
ultramarathon
ultramarathon
half marathon
running
marathon
inline speed skating
heptathlon
heptathlon
decathlon
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
demolition derby
demolition derby
monster truck
mud bogging
motocross
grand prix motorcycle racing
telemark skiing
snowboarding
telemark skiing
nordic skiing
ski touring
skijoring
whitewater kayaking
whitewater kayaking
rafting
kayaking
canoeing
adventure racing
arena football
indoor american football
arena football
canadian football
american football
women's lacrosse
reining
barrel racing
rodeo
reining
cowboy action shooting
bull riding
eight-ball
nine-ball
blackball (pool)
trick shot
eight-ball
straight pool

Video Classification - Datasets

- YouTube-8M
  - millions of videos labeled with thousands of classes,
  - a large-scale benchmark dataset for general multi-label video classification.
  - 8 million videos
  - 500K hours of videos annotated with a vocabulary of 4800 visual entities.



http://research.google.com/youtube8m/

Video Classification - Datasets

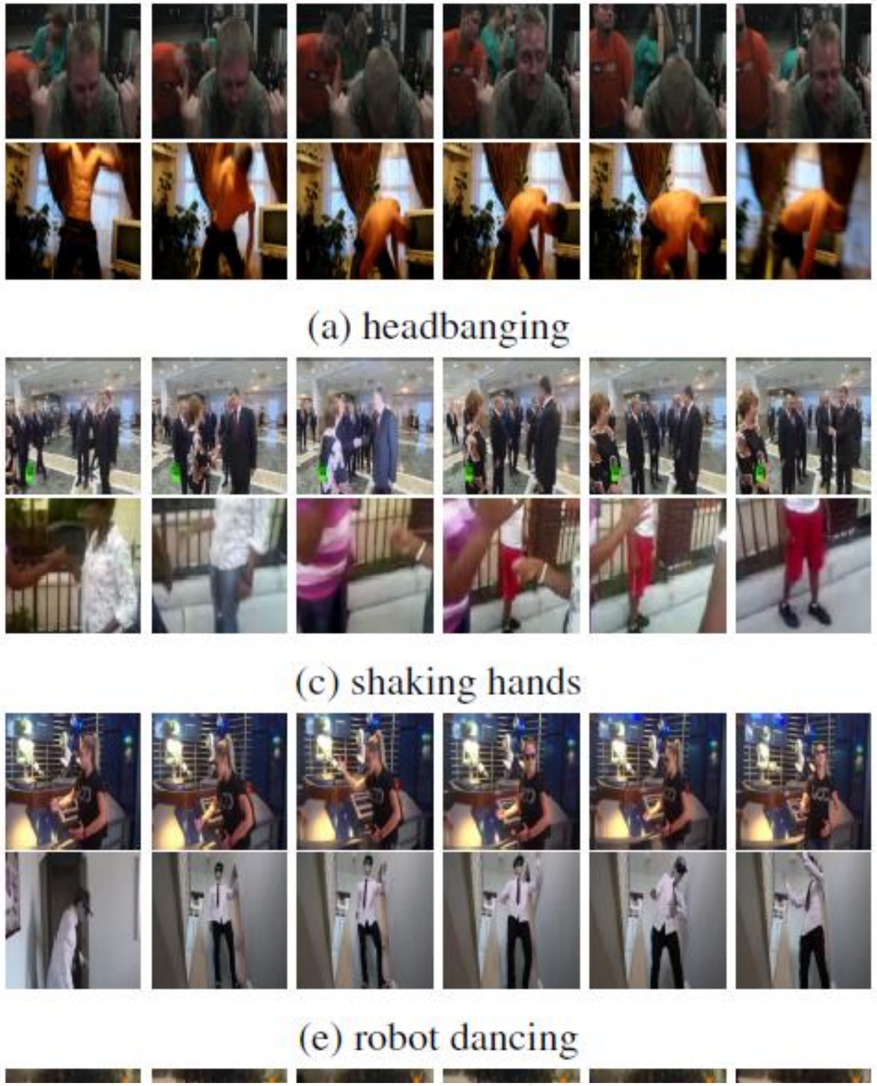
- YouTube-8M

Accordion Action-adventure-game Advertising Aircraft Airport Album American-football Amusement-park
Animation Art Ballet Baseball Basketball Batman Battlefield Bicycle Bird BMW Boat Bollywood Book Bowling Boxing Call-of-Duty
Camera Car Carnival Cartoon Cat Cheerleading Choir Christmas Circus Clash-of-Clans Climbing Combat Comedy Comic-book Comics Computer
Concert Cooking Cooking-show Cosmetics Counter-Strike Cricket Cue-sports Cycling Dance Dashcam Disc-jockey
Diving Dog Doll Dragon-Ball Drawing Drifting Drums DVD Earth Engine Fashion Festival Figure-skating Final-Fantasy Fishing Flamenco Food Football
Games Gardening Grand-Theft-Auto-IV Grand-Theft-Auto-V Guitar Gymnastics Hair Hairstyle Halo Handball Handheld-game-console High-school
Highlight-film Hockey Home-improvement Horse Horse-racing Hotel House Human-swimming Hunting Ice-skating iPad iPhone iPod Kayak Knife
Landing Laptop League-of-Legends LEGO Medicine Microsoft-Windows Minecraft Mixtape Mobile-phone Model-aircraft Motocross Motorcycle
Motorsport Mountain-bike Music-video Musical-ensemble Nail Naruto Nature News-program Orchestra Origami
Outdoor-recreation Painting Parachuting Personal-computer Photography Piano Pokémon Pool Prayer Racing Radio-controlled-aircraft Radio-controlled-car
Radio-controlled-model Rallying Recipe Roller-skating Rugby-football RuneScape Running Samsung-Galaxy School Shoe Simulation-video-game Sitcom
Skateboarding Sketch-comedy Skiing Sledding Slide-show Smartphone Snowboarding Sonic-the-Hedgehog Sports-game Star-Wars
Strategy-video-game Surfing Tablet-computer Talent-show Tank Television Television-advertisement Tennis The-Sims The-Walt-Disney-Company Touhou-Project
Toy Tractor Tractor-pulling Trailer Train Truck University Vampire Vehicle Video-game Video-game-console Violin
Warcraft Water Weapon Weather Wedding Weight-training Wii Winter-sport Woodturning World-of-Warcraft Wrestling Xbox

http://research.google.com/youtube8m/

Video Classification - Datasets

- Kinetics
  - Youtube videos
  - 650,000 video clips covering 400/600/700 human action classes
  - >= 600 video clips for each action class.
  - Each video clip lasts around 10 seconds and is labeled with a single action class



https://github.com/rocksyne/kinetics-dataset-downloader sensetime

Tsinghua University

Outline

Video classification - challenges

Challenges:

- Computationally expensive
- Lower quality:

  Resolution, motion blur, occlusion
- Requires lots of training data!

Balance the efficiency and performance!

A video model need to:

- Sequence modeling
- Temporal reasoning (receptive field)

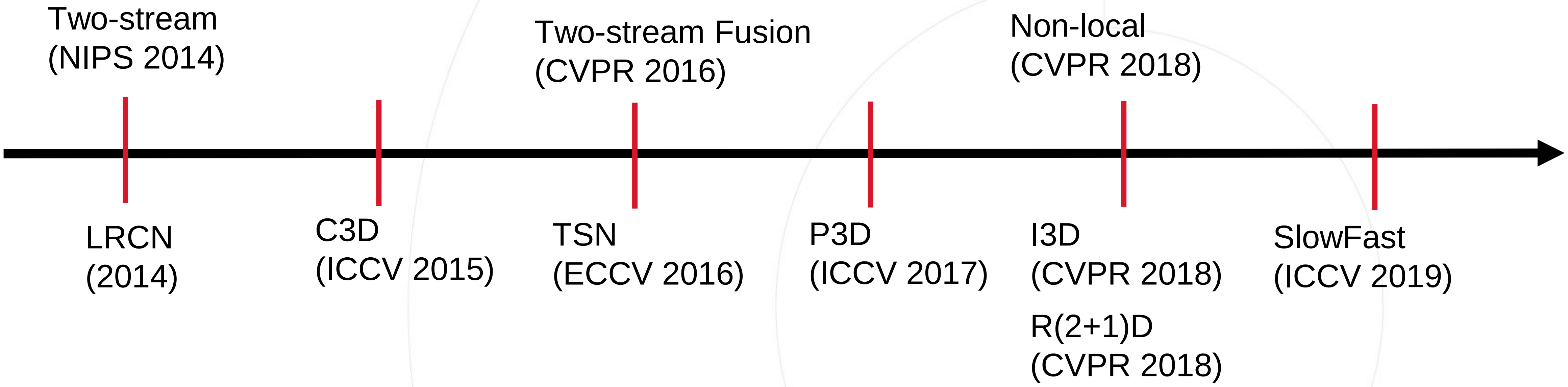
Input video:

$$T \times 3 \times H \times W$$

Videos are ~30 frames per second (fps)

Size of uncompressed video:
3 bytes per pixel
SD (640x480): ~1.5GB per minute
HD (1920x1080): ~10GB per minute

Video classification



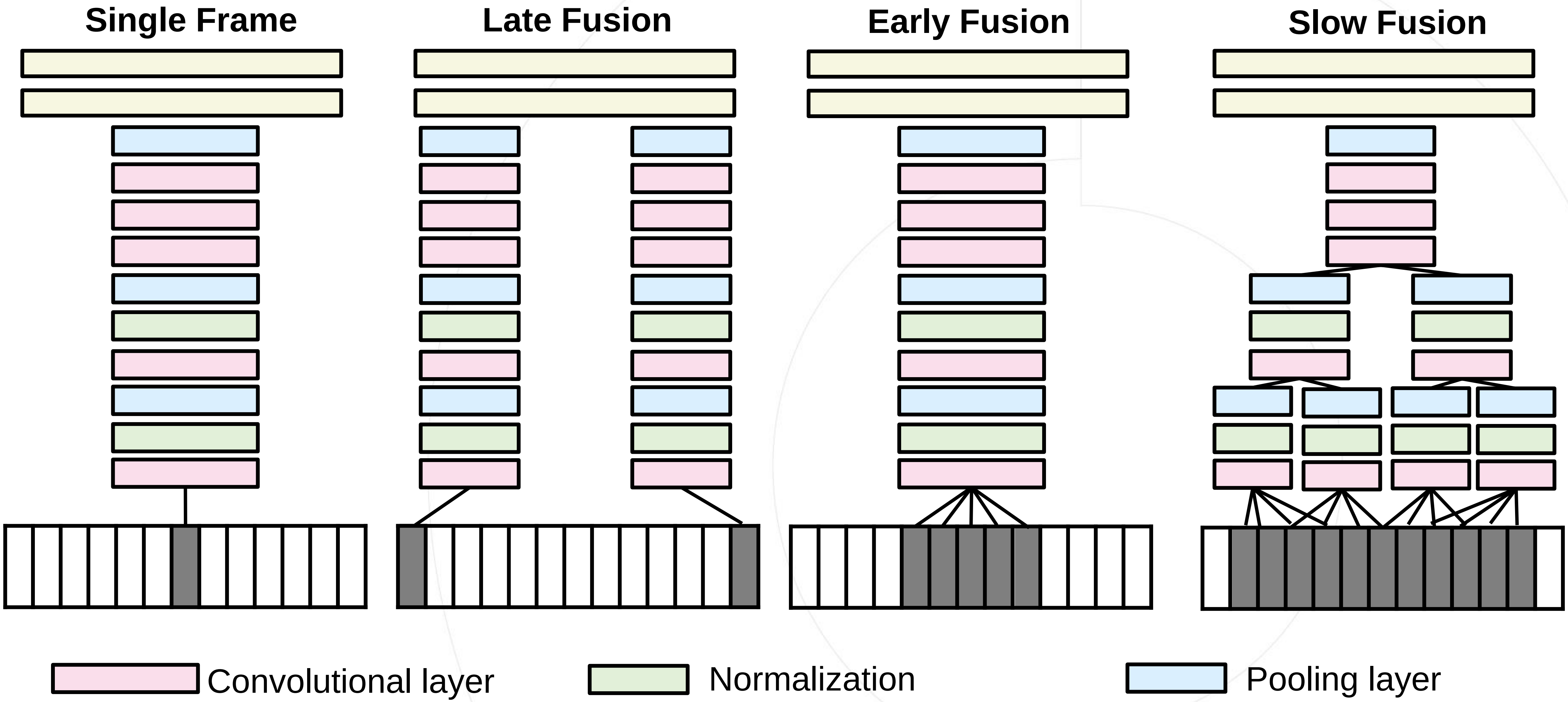
Raw video: Long, with high FPS (frames per second)

Training: Train model to classify short clips with low FPS (frames per second)

Testing: Run model on different clips, average predictions

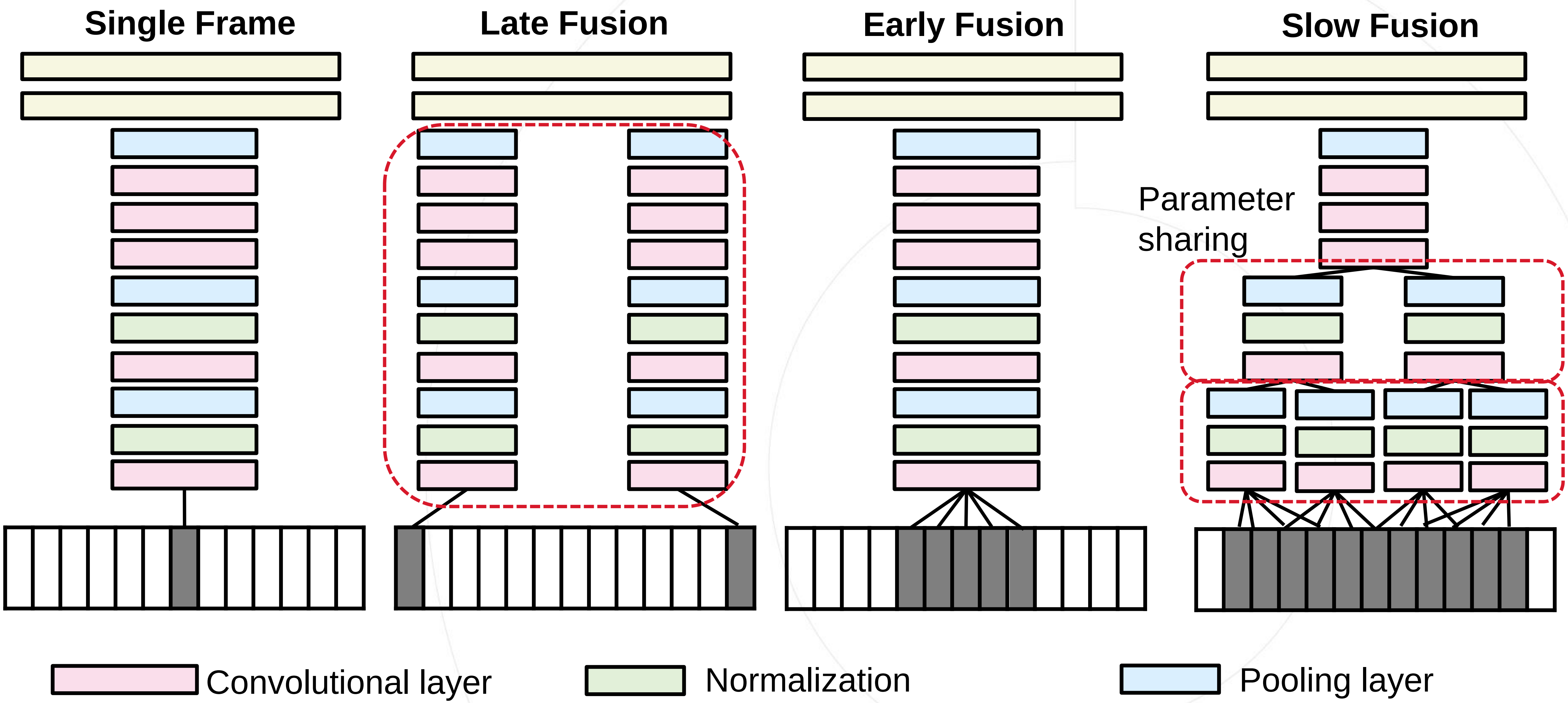
Slide credit: Justin Johnson

Video classification - pipelines



A. Karpathy et al. Large-scale Video Classification with Convolutional Neural Networks In CVPR 2014.

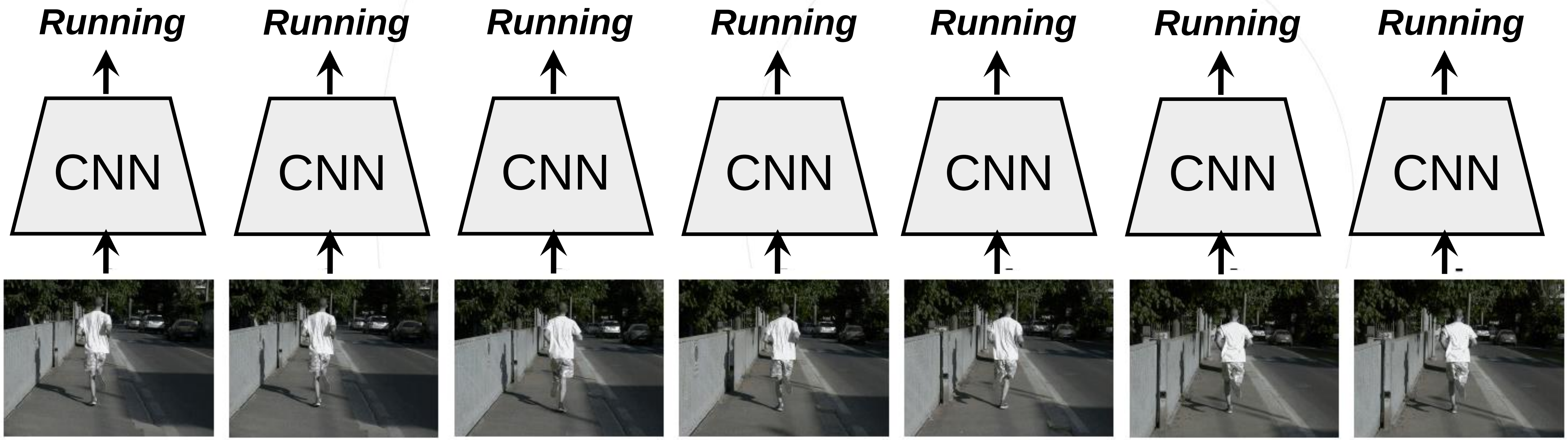
Video classification - pipelines



A. Karpathy et al. Large-scale Video Classification with Convolutional Neural Networks In CVPR 2014.

Video Classification: Single-Frame CNN

- Training: train normal 2D CNN to classify video frames independently
- Testing: average predicted probabilities
- Simple, yet strong baseline for video classification!



A. Karpathy et al. Large-scale Video Classification with Convolutional Neural Networks In CVPR 2014.
Based on Slides of Justin Johnson

Video Classification: Early Fusion

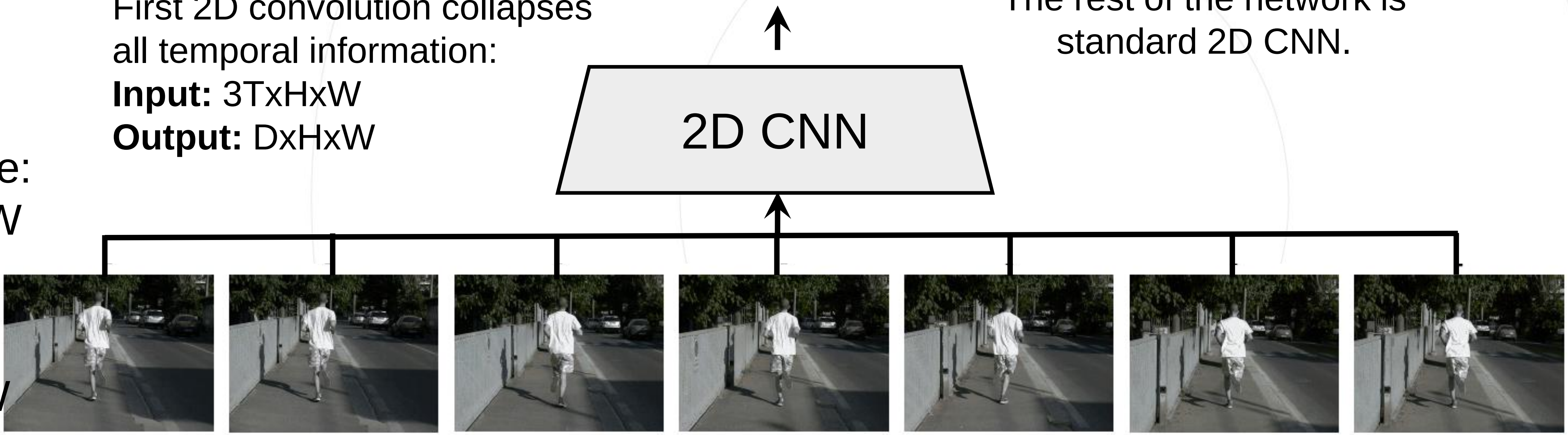
- Compare frames with very first conv layer, after that normal 2D CNN
- Combine information across an entire time window immediately on the pixel level. The early and direct connectivity to pixel data allows the network to precisely detect local motion direction and speed.

Class score: C

First 2D convolution collapses all temporal information:
Input: 3TxHxW
Output: DxHxW

The rest of the network is standard 2D CNN.

Reshape:
3TxHxW

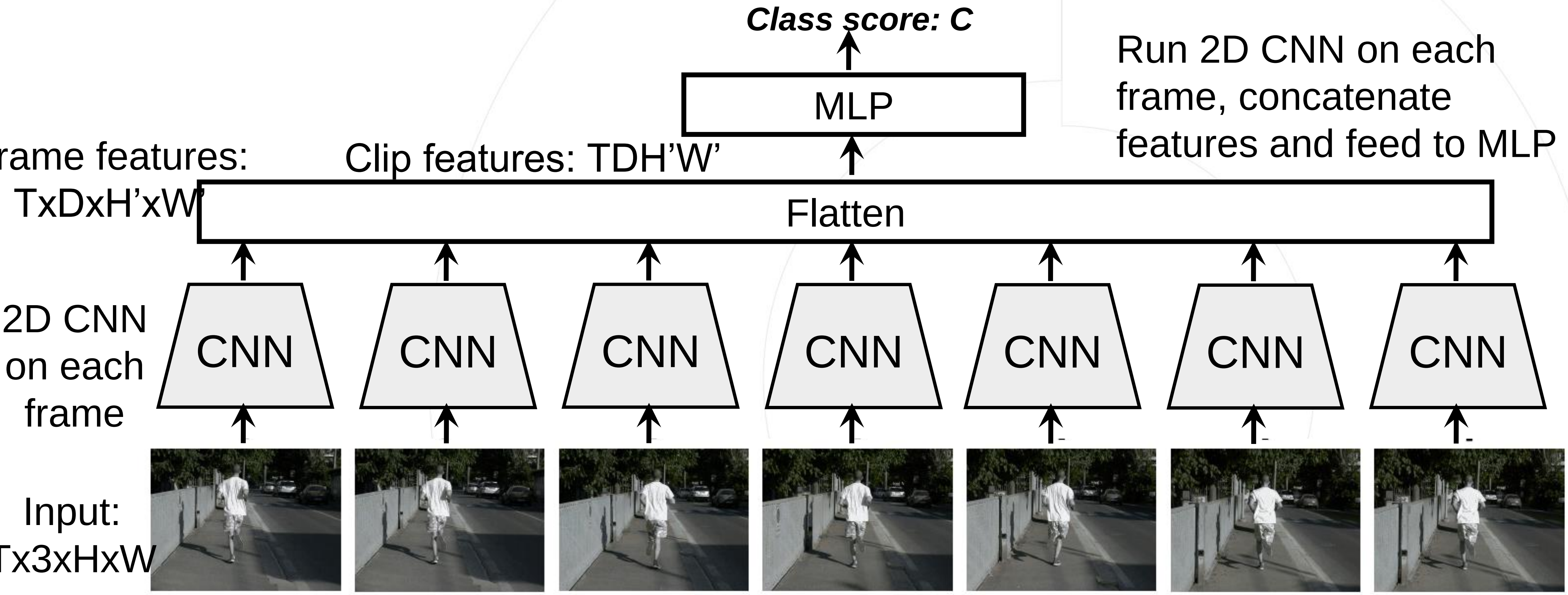


Input:
Tx3xHxW

A. Karpathy et al. Large-scale Video Classification with Convolutional Neural Networks In CVPR 2014.
Based on Slides of Justin Johnson

Video Classification: Late Fusion (with FC layers)

- Get high-level appearance of each frame, combine them together

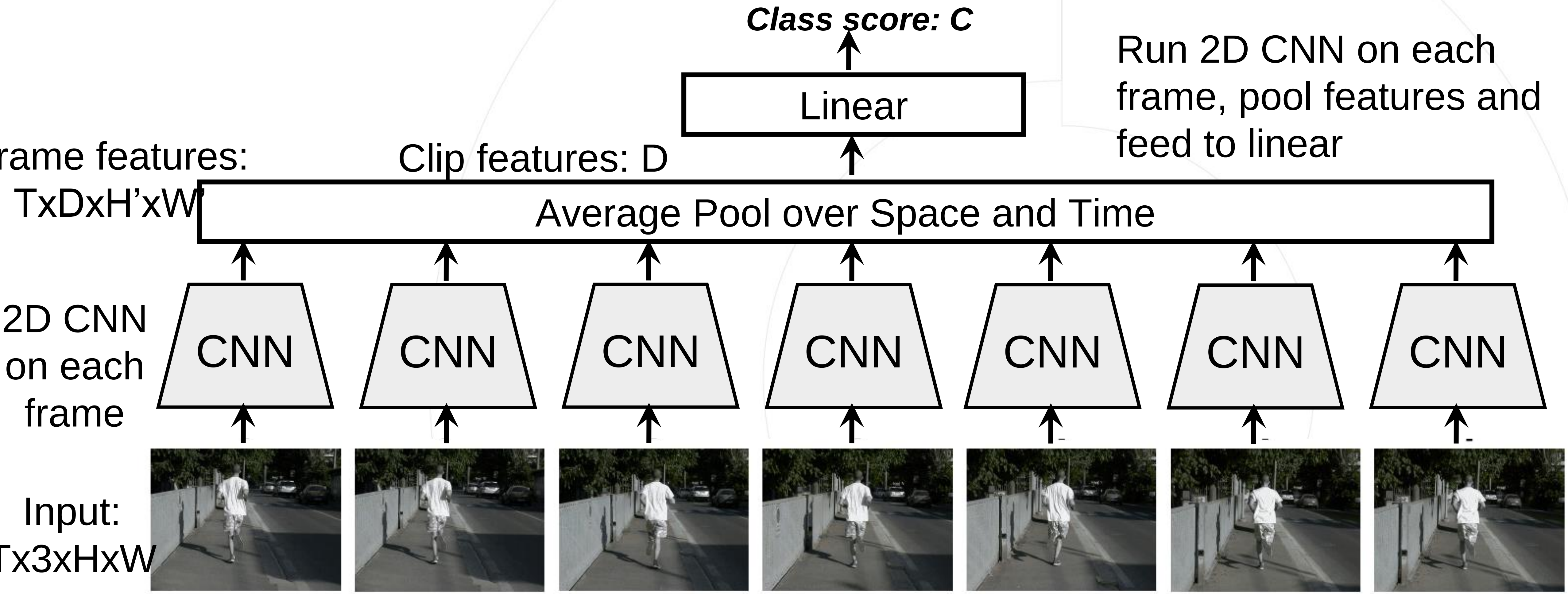


Run 2D CNN on each frame, concatenate features and feed to MLP

Tsinghua University sensetime

Video Classification: Late Fusion (with pooling layers)

- Get high-level appearance of each frame, combine them together

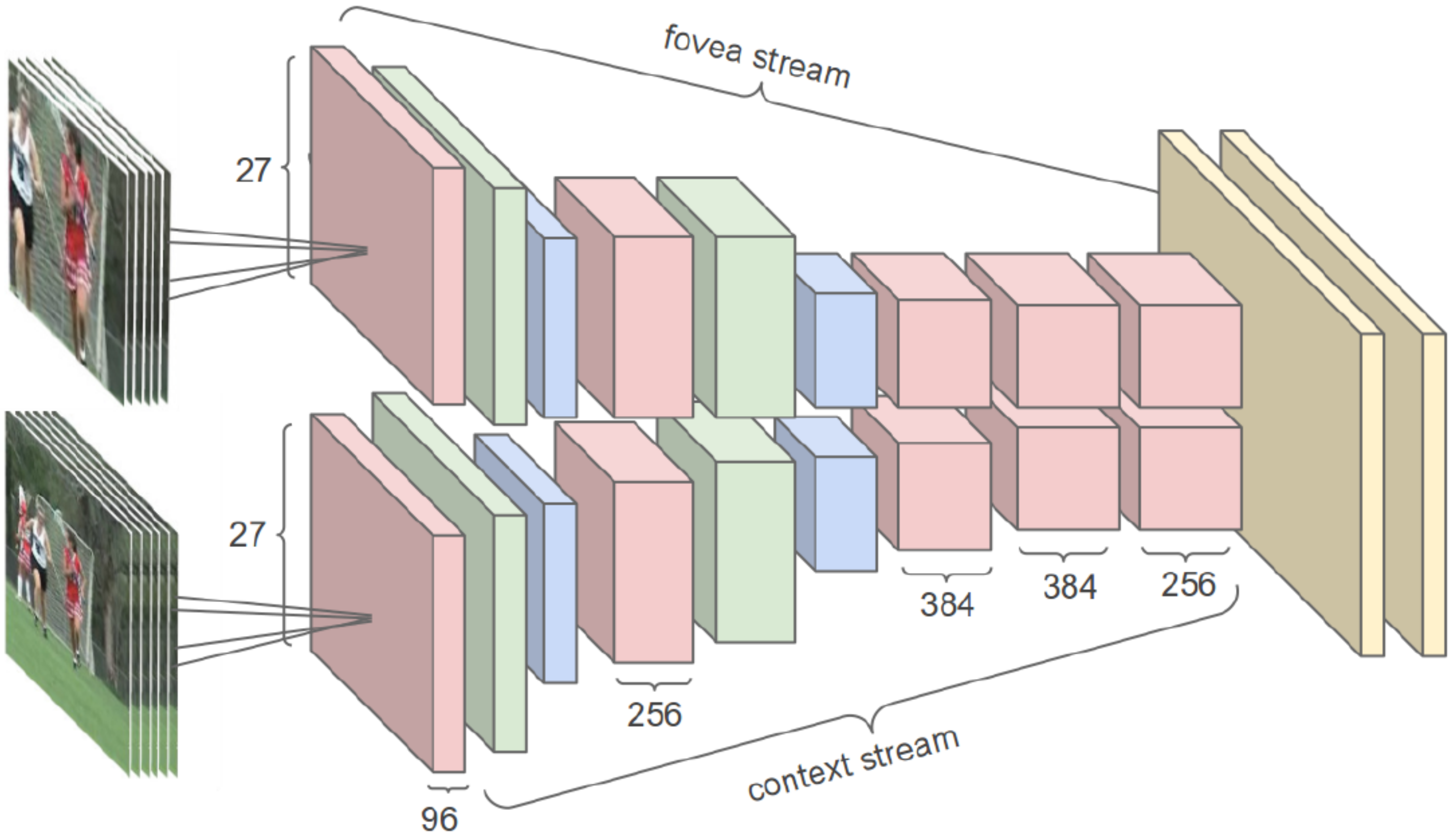


A. Karpathy et al. Large-scale Video Classification with Convolutional Neural Networks In CVPR 2014.
Based on Slides of Justin Johnson

Video classification: Multiresolution CNNs

Fovea stream: receives the center 89x89 region at the original resolution.

Context stream: receives the down-sampled frames at half the original spatial resolution (89x89 pixels)



A. Karpathy et al. Large-scale Video Classification with Convolutional Neural Networks In CVPR 2014.

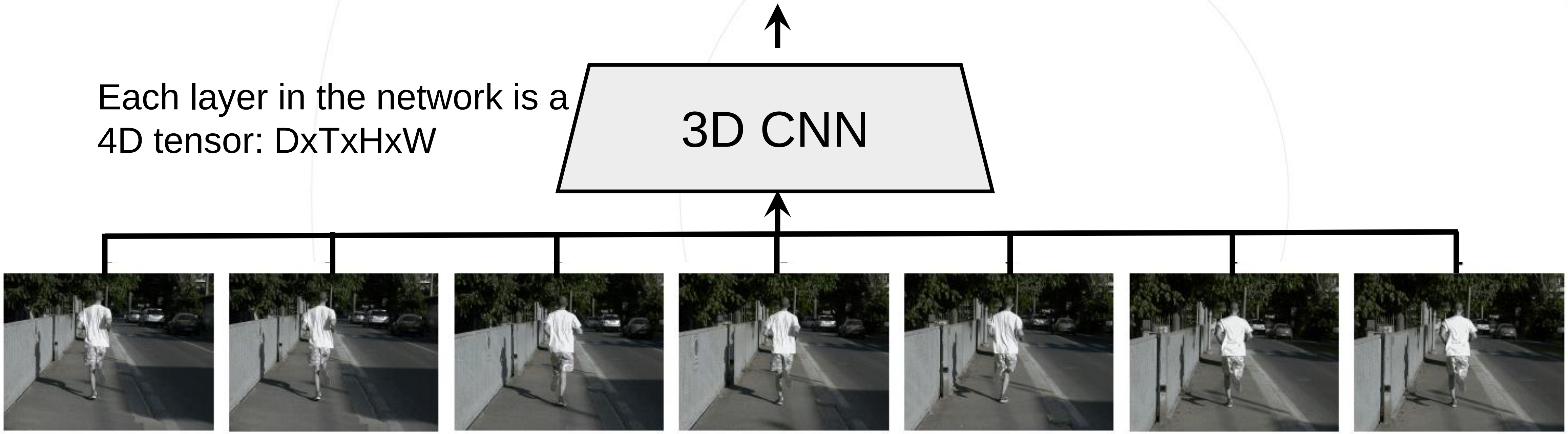
Video classification: Performance comparison

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.

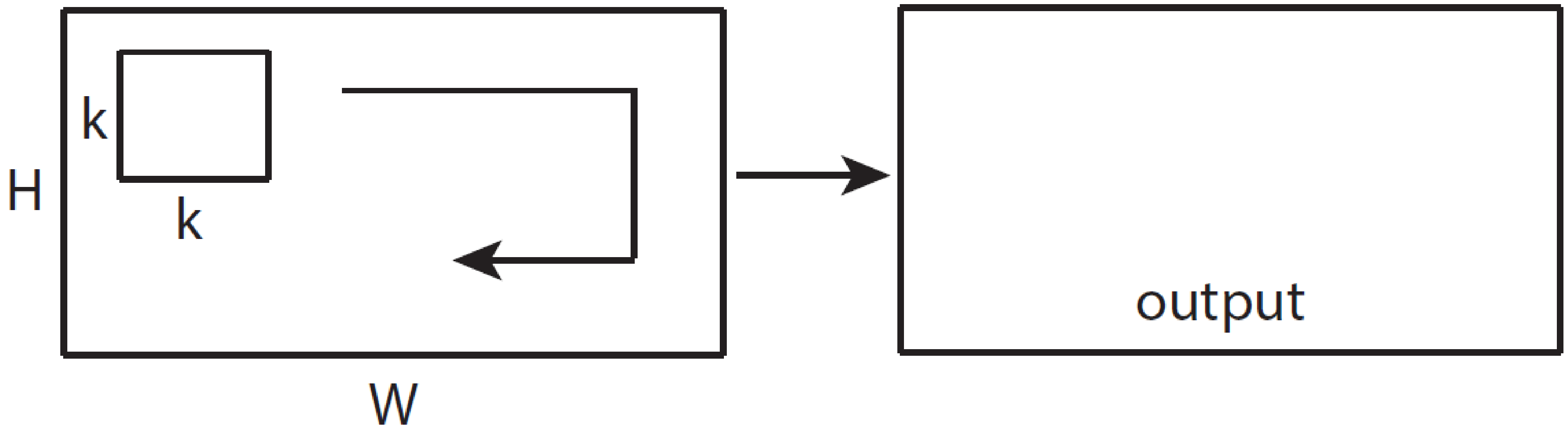
Video Classification: 3D CNN

- Use 3D versions of convolution and pooling to slowly fuse temporal information over the course of the network



A. Karpathy et al. Large-scale Video Classification with Convolutional Neural Networks In CVPR 2014.
Based on Slides of Justin Johnson

Video Classification: 3D CNN – C3D



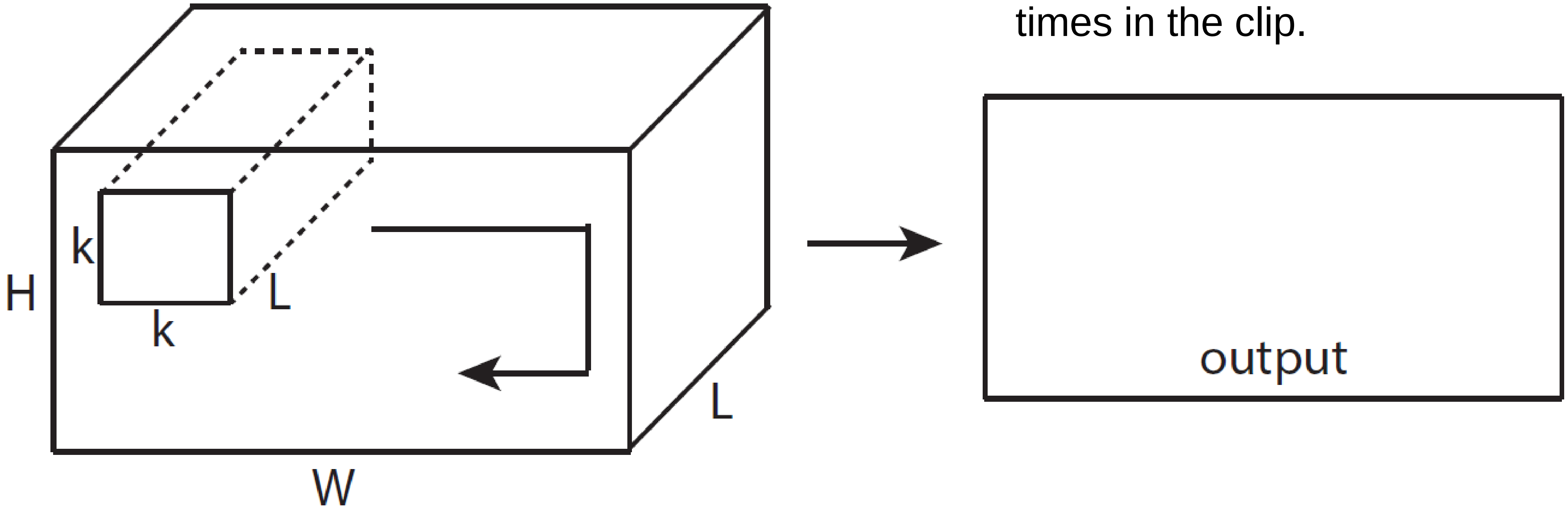
(a) 2D convolution

Sliding over x, y

Du Tran, et al. Learning spatiotemporal features with 3D convolutional networks. In Proc. ICCV, 2015.

Video Classification: 3D CNN

No temporal shift-invariance! The convolutional filters are different for the same motion at different times in the clip.

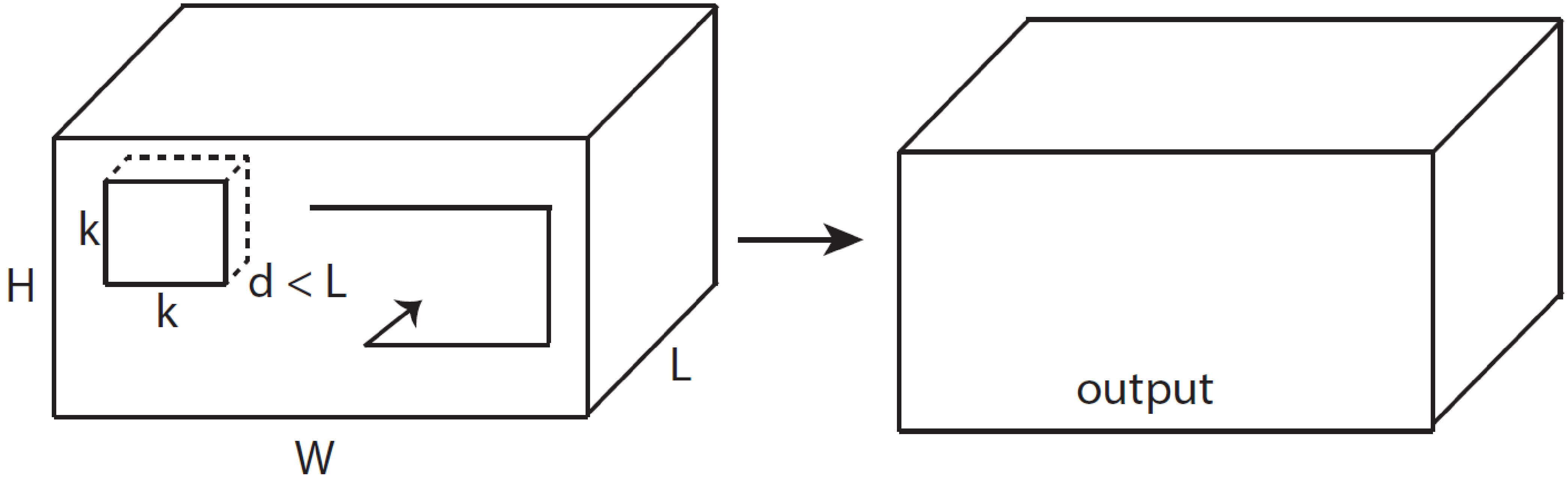


(b) 2D convolution on multiple frames

Du Tran, et al. Learning spatiotemporal features with 3D convolutional networks. In Proc. ICCV, 2015.

Video Classification: 3D CNN

Temporal shift-invariance.
Each convolutional filter slides over time, x, y.

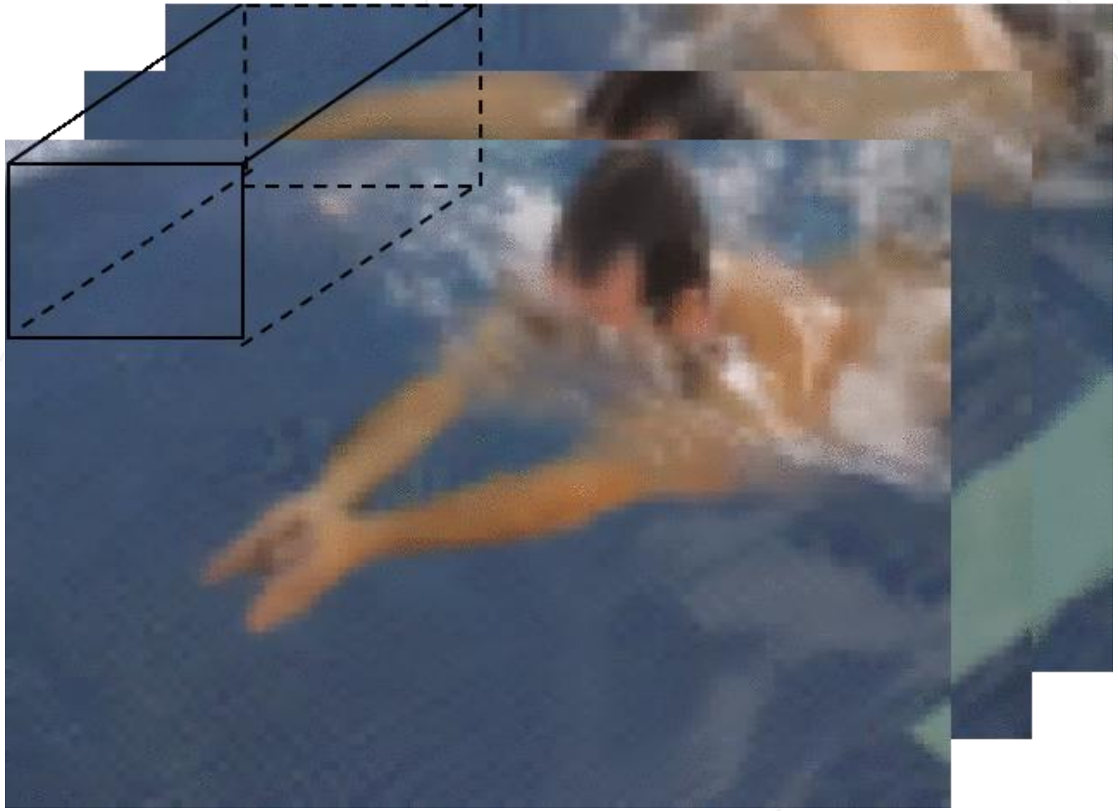


(c) 3D convolution

Du Tran, et al. Learning spatiotemporal features with 3D convolutional networks. In Proc. ICCV, 2015.

Video Classification: 3D CNN – C3D

- VGG of 3D CNNs
- Uses all 3x3x3 conv and 2x2x2 pooling (except Pool1)
- Release pre-trained model on Sports-1M: can be used as a video feature extractor



Du Tran, et al. Learning spatiotemporal features with 3D convolutional networks. In Proc. ICCV, 2015.

Video classification: 3D CNN

Video Classification: 3D CNN – C3D

- VGG of 3D CNNs
- Uses all 3x3x3 conv and 2x2x2 pooling (except Pool1)
- Release pre-trained model on Sports-1M: can be used as a video feature extractor

- Problem: 3D conv is very expensive!
- AlexNet: 0.7 GFLOP
- VGG-16: 13.6 GFLOP
- **C3D: 39.5 GFLOP**

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

Du Tran, et al. Learning spatiotemporal features with 3D convolutional networks. In Proc. ICCV, 2015.
Based on Slides of Justin Johnson

Video Classification: 3D CNN – C3D

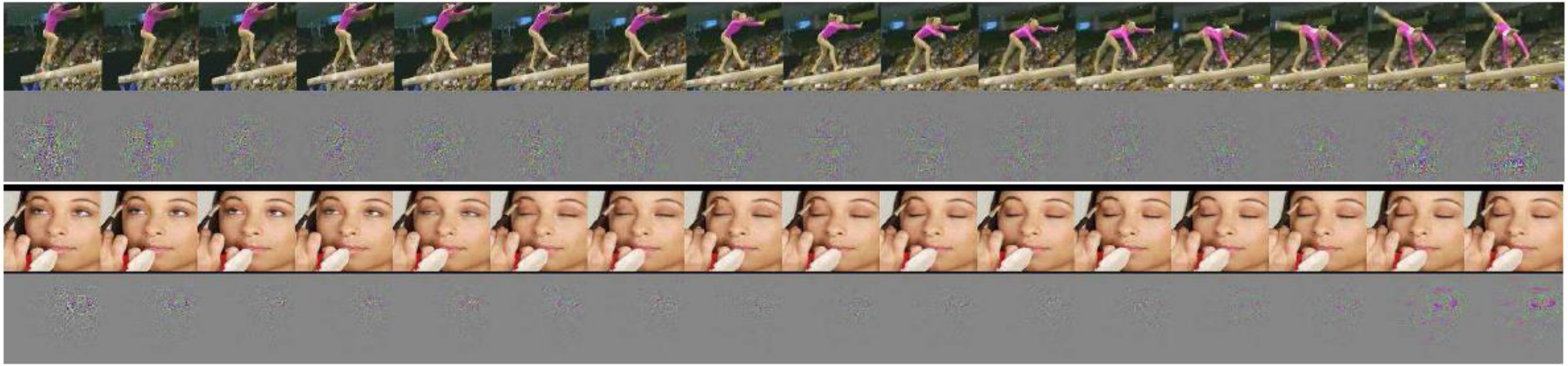


Figure 4. **Visualization of C3D model, using the method from [46].** Interestingly, C3D captures appearance for the first few frames but thereafter only attends to salient motion. Best viewed on a color screen.

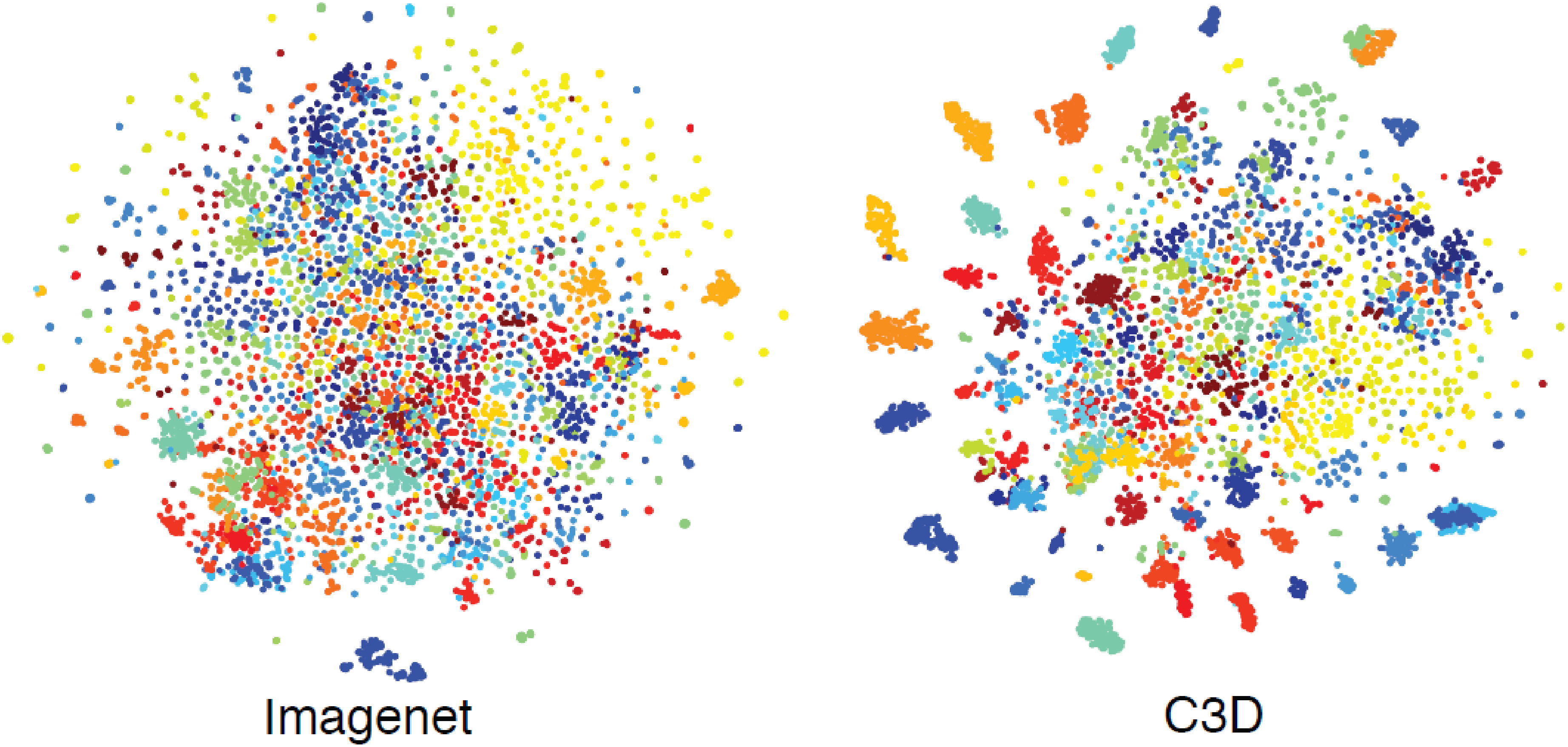
Du Tran, et al. Learning spatiotemporal features with 3D convolutional networks. In Proc. ICCV, 2015.

Video Classification: 3D CNN – C3D

https://vlg.cs.dartmouth.edu/c3d/

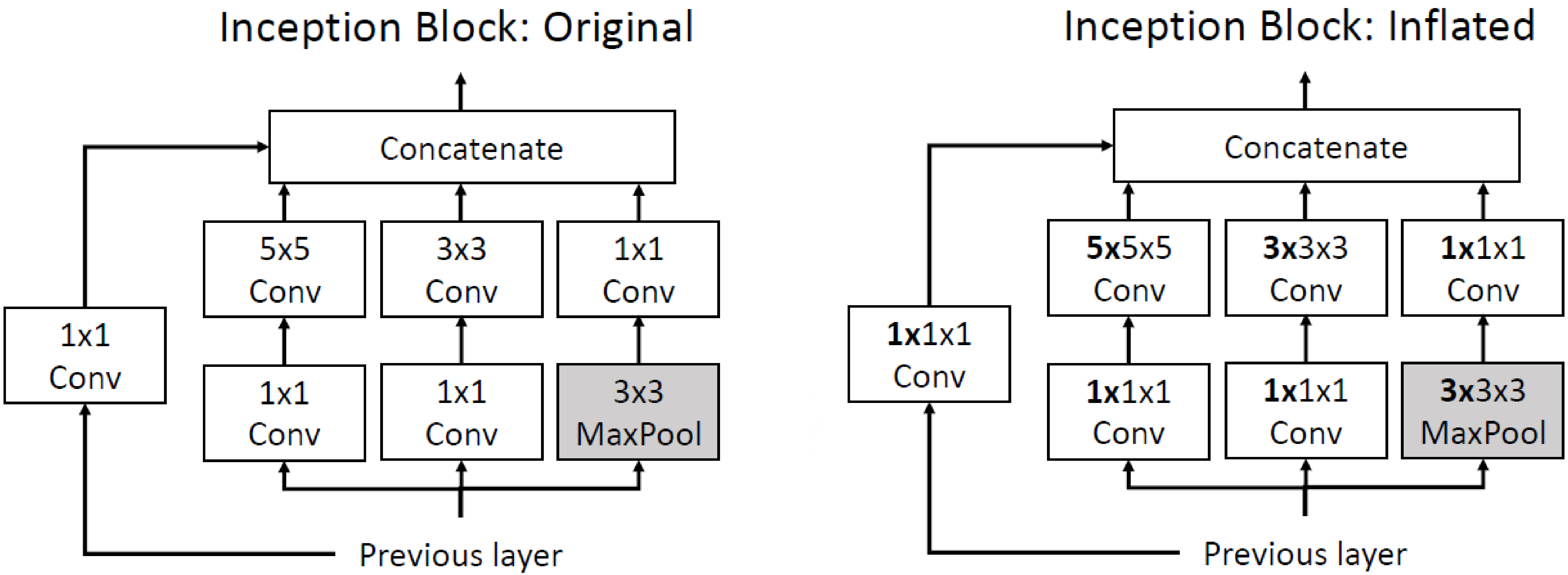
Du Tran, et al. Learning spatiotemporal features with 3D convolutional networks. In Proc. ICCV, 2015.

Video Classification: 3D CNN – C3D



Du Tran, et al. Learning spatiotemporal features with 3D convolutional networks. In Proc. ICCV, 2015.

Video Classification: 3D CNN – I3D

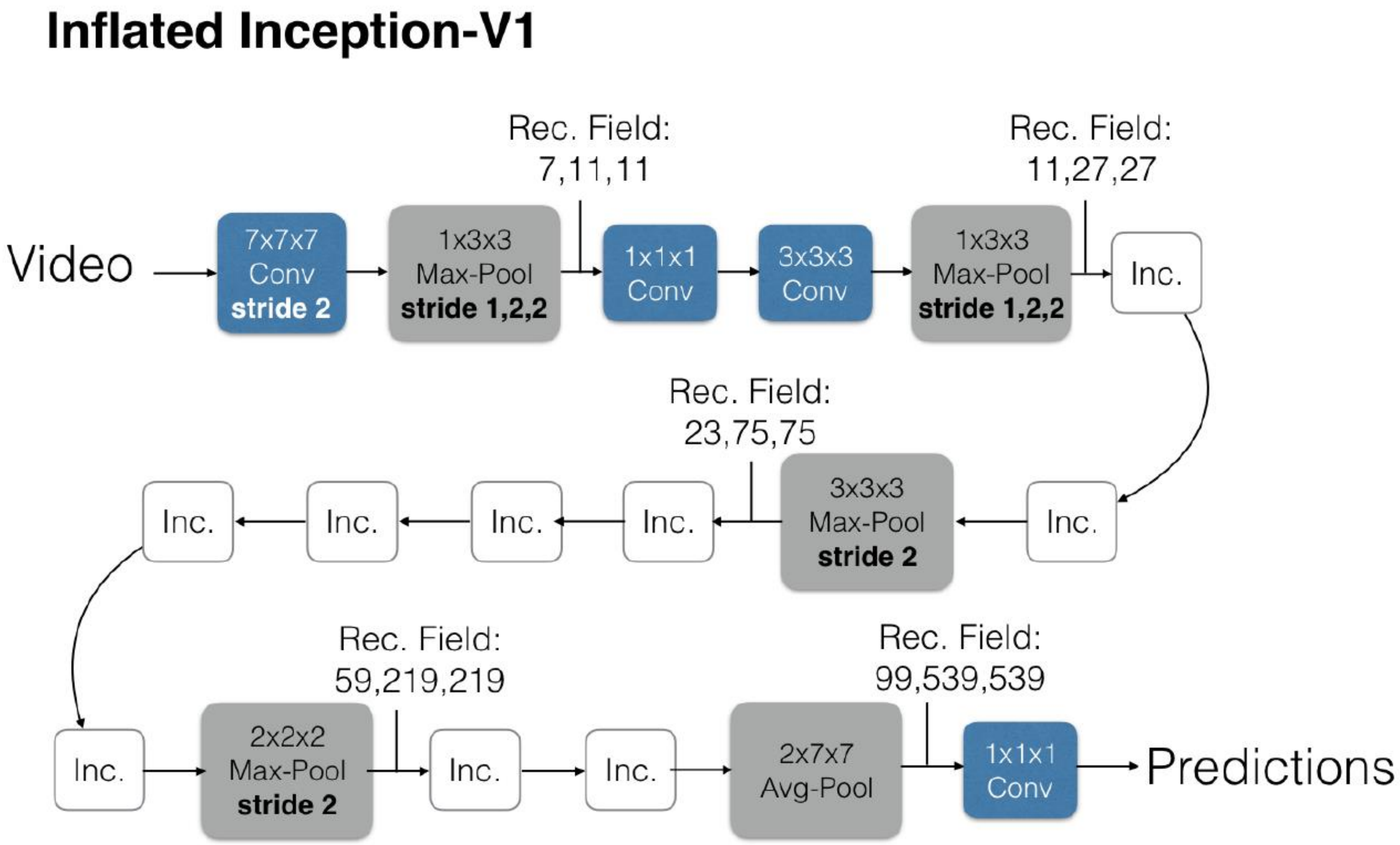


Carreira et al.. Quo Vadis, Action Recognition? A New Model and the Kinetics Dataset. In Proc. CVPR , 2017.

Video Classification: 3D CNN – I3D

- **Inflating 2D ConvNets into 3D.** Make square filters cubic – $N \times N$ filters become $N \times N \times N$.

- Use ImageNet-pretrained Inception-V1 as base network. Repeat the 2D pre-trained weights in the 3rd dimension



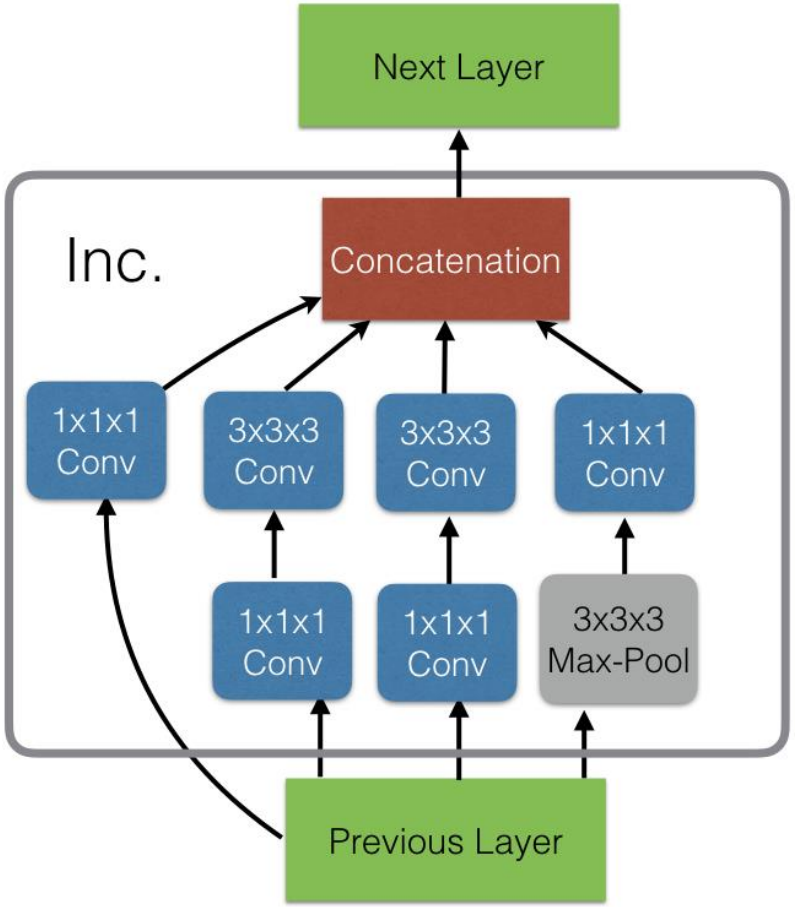
Carreira et al.. Quo Vadis, Action Recognition? A New Model and the Kinetics Dataset. In Proc. CVPR , 2017.

Video Classification: 3D CNN – I3D

- **Inflating 2D ConvNets into 3D.** Make square filters cubic – $N \times N$ filters become $N \times N \times N$.

- Use ImageNet-pretrained Inception-V1 as base network. Repeat the 2D pre-trained weights in the 3rd dimension

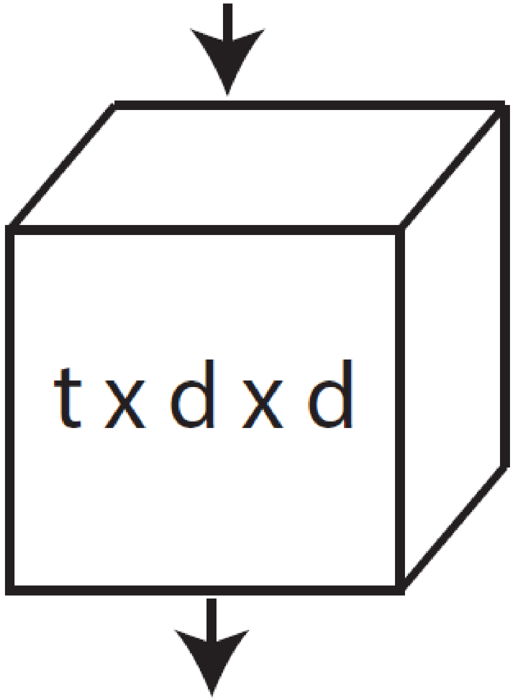
Inception Module (Inc.)



Carreira et al.. Quo Vadis, Action Recognition? A New Model and the Kinetics Dataset. In Proc. CVPR , 2017.

Video classification: 3D CNN

Video Classification: 3D CNN – R(2+1)D



3D conv:

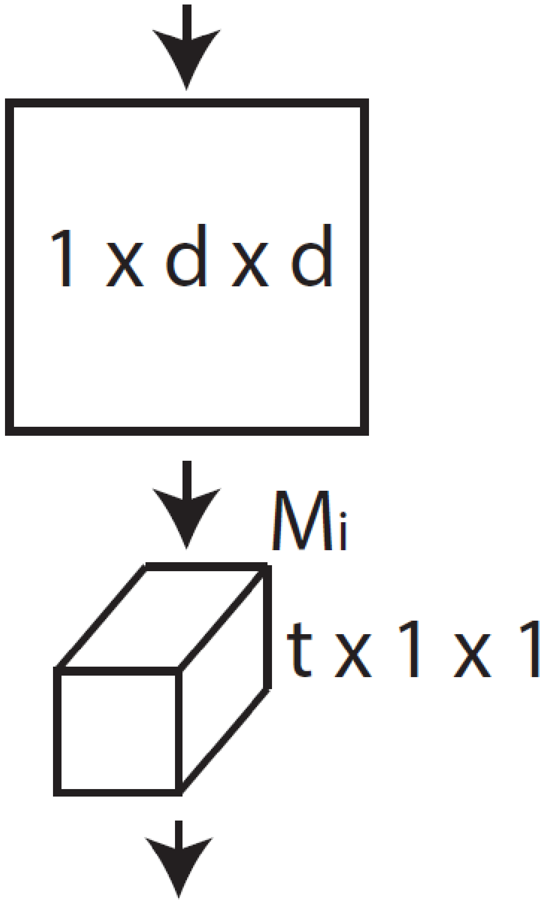
$$C_{out} \times C_{in} \times t \times d \times d$$

Full 3D convolution is carried out using a filter of size t×d×d where t denotes the temporal extent and d is the spatial width and height.

Du Tran, et al. A Closer Look at Spatiotemporal Convolutions for Action Recognition. *In Proc. CVPR, 2018.*

Video classification: 3D CNN

Video Classification: 3D CNN – R(2+1)D



b)

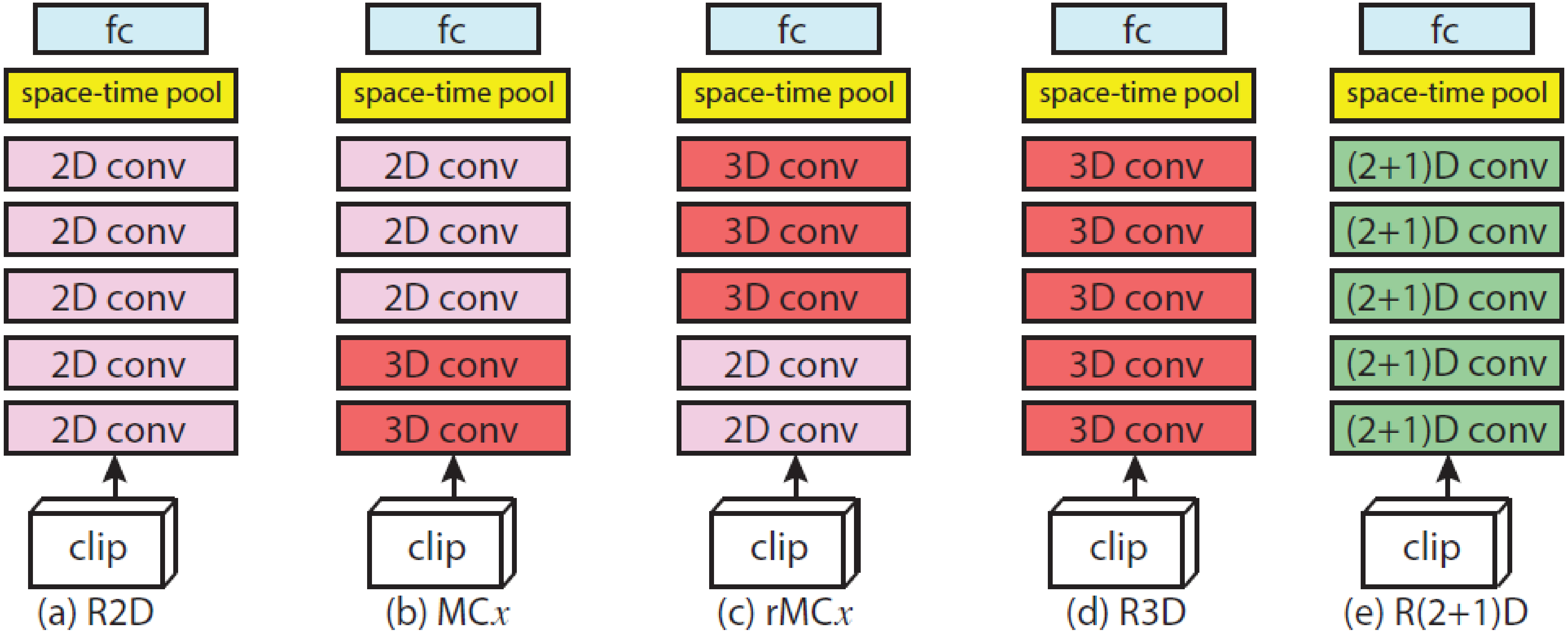
(2+1)D conv:

$$C_{out} \times M \times 1 \times d \times d + M \times C_{in} \times t \times 1 \times 1$$

A (2+1)D convolutional block splits the computation into a spatial 2D convolution followed by a temporal 1D convolution. The numbers of 2D filters (M) are hyper-parameters.

Du Tran, et al. A Closer Look at Spatiotemporal Convolutions for Action Recognition. *In Proc. CVPR, 2018.*

Video Classification: 3D CNN – R(2+1)D



Du Tran, et al. A Closer Look at Spatiotemporal Convolutions for Action Recognition. *In Proc. CVPR, 2018.*

Video classification: 3D CNN – performance comparison

method	Clip@1	Video@1	Video@5
DeepVideo [16]	41.9	60.9	80.2
C3D [36]	46.1	61.1	85.2
2D Resnet-152 [13]	46.5*	64.6*	86.4*
Conv pooling [42]	-	71.7	90.4
P3D [25]	47.9*	66.4*	87.4*
R3D-RGB-8frame	53.8	-	-
R(2+1)D-RGB-8frame	56.1	72.0	91.2
R(2+1)D-Flow-8frame	44.5	65.5	87.2
R(2+1)D-Two-Stream-8frame	-	72.2	91.4
R(2+1)D-RGB-32frame	**57.0**	**73.0**	**91.5**
R(2+1)D-Flow-32frame	46.4	68.4	88.7
R(2+1)D-Two-Stream-32frame	-	**73.3**	**91.9**

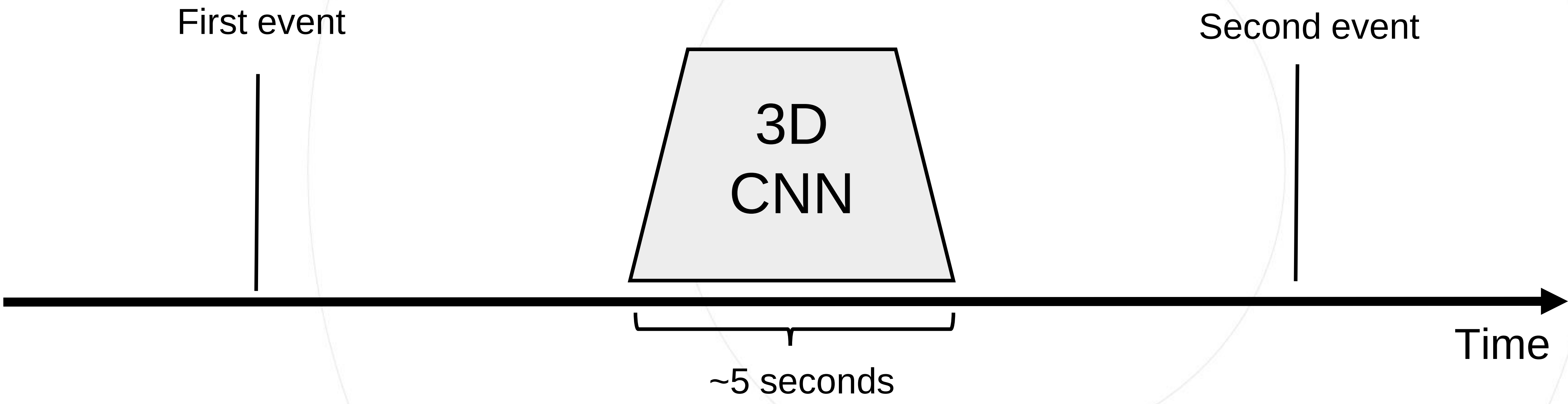
Table 4. **Comparison with the state-of-the-art on Sports-1M.** R(2+1)D outperforms C3D by 10.9%, and P3D by 9.1% and it achieves the best reported accuracy on this benchmark to date. *These baseline numbers are taken from [25].

method	pretraining dataset	top1	top5
I3D-RGB [4]	none	67.5	87.2
I3D-RGB [4]	ImageNet	72.1	90.3
I3D-Flow [4]	ImageNet	65.3	86.2
I3D-Two-Stream [4]	ImageNet	**75.7**	**92.0**
R(2+1)D-RGB	none	72.0	90.0
R(2+1)D-Flow	none	67.5	87.2
R(2+1)D-Two-Stream	none	73.9	90.9
R(2+1)D-RGB	Sports-1M	74.3	91.4
R(2+1)D-Flow	Sports-1M	68.5	88.1
R(2+1)D-Two-Stream	Sports-1M	**75.4**	**91.9**

Table 5. **Comparison with the state-of-the-art on Kinetics.** R(2+1)D outperforms I3D by 4.5% when trained from scratch on RGB. R(2+1)D pretrained on Sports-1M outperforms I3D pretrained on ImageNet, for both RGB and optical flow. However, it is slightly worse than I3D (0.3%) when fusing the two streams.

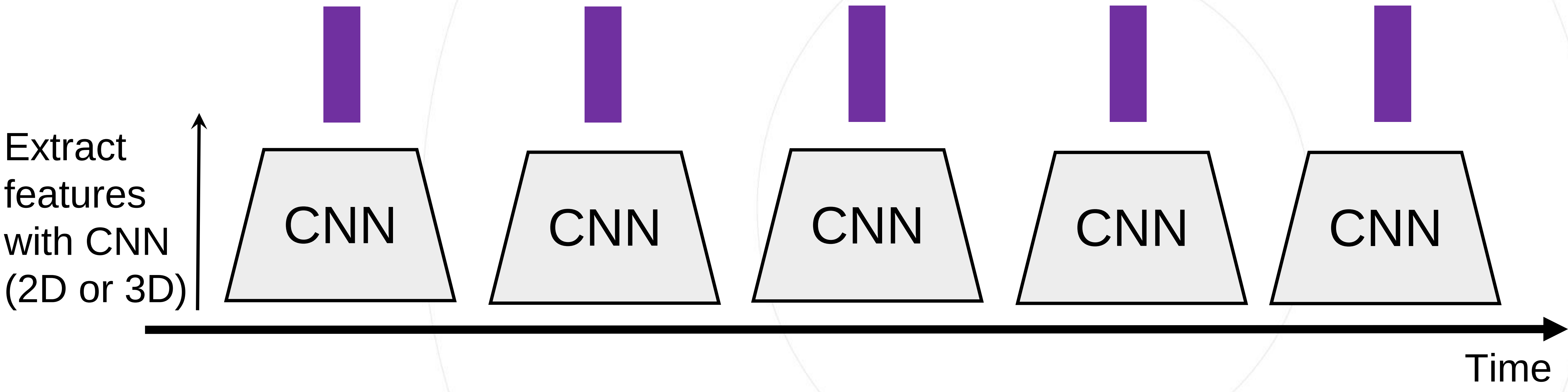
Video classification: modeling long-term temporal structure

Temporal CNNs can only model local motion between frames in very short clips of ~2-5 seconds. How to model long-term structure is a remaining issue.

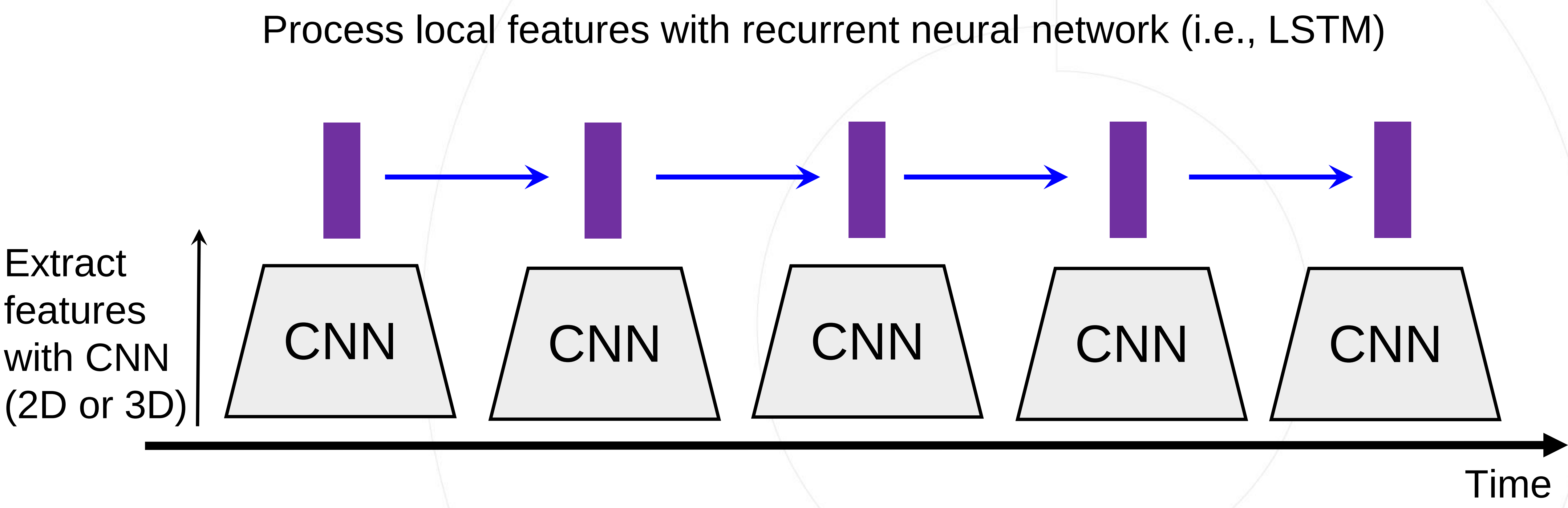


Based on Slides of Justin Johnson

Video classification: modeling long-term temporal structure



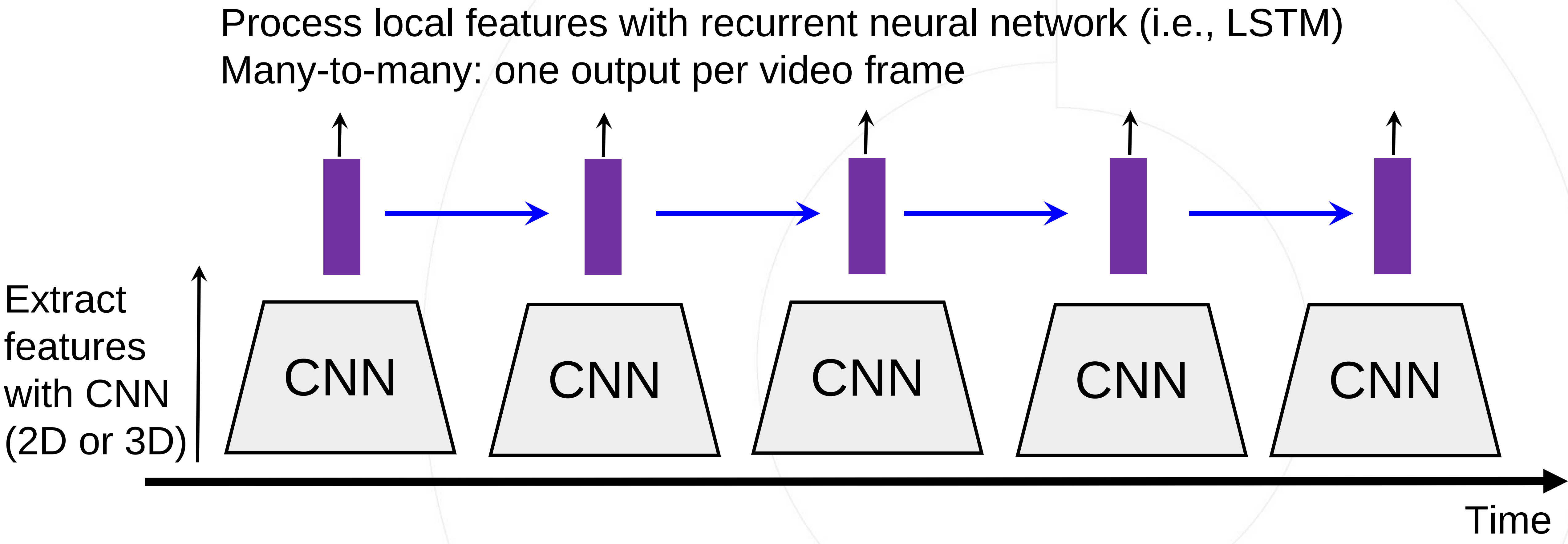
Video classification: modeling long-term temporal structure



Video classification: CNN+RNN/LSTM

Tsinghua University sensetime

Video classification: modeling long-term temporal structure

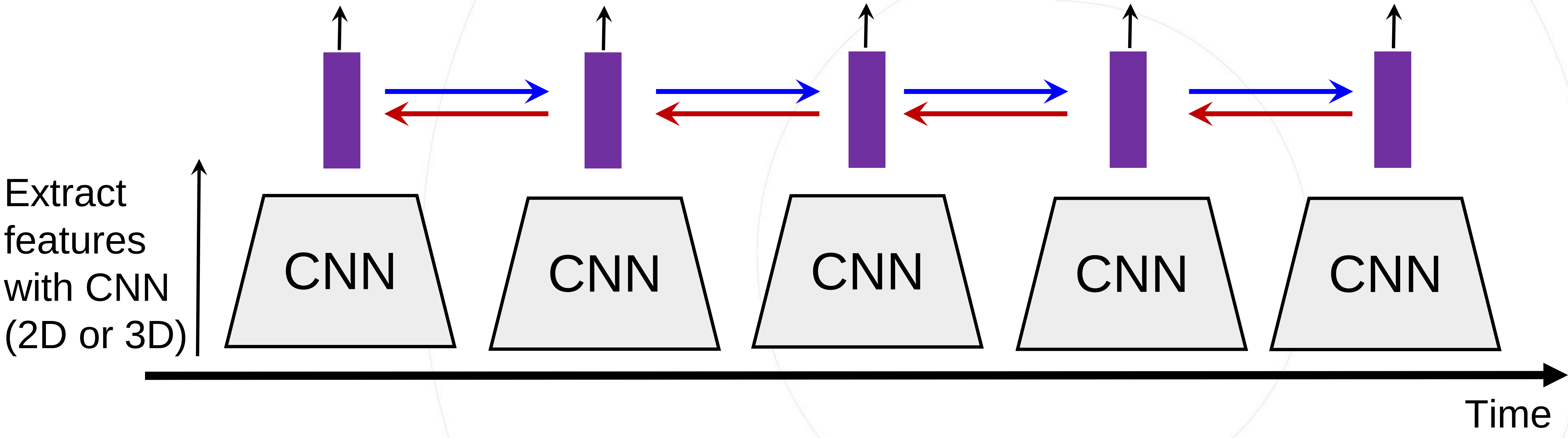


Based on Slides of Justin Johnson

Video classification: CNN+RNN/LSTM

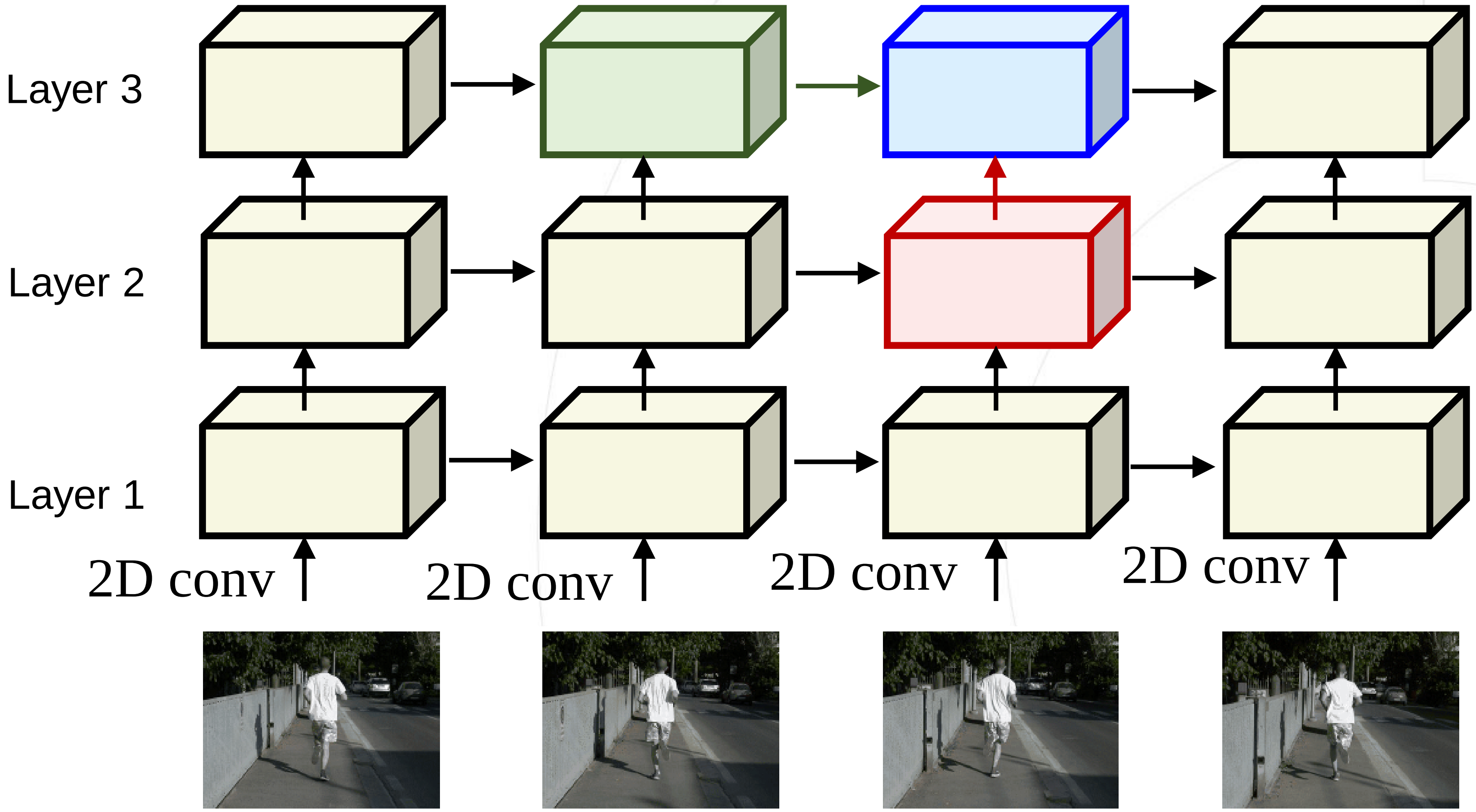
Video classification: modeling long-term temporal structure

Sometimes use the pre-trained CNN as the feature extractor,
Don't backpropapagte to CNN to save memory.



Based on Slides of Justin Johnson

Recurrent Neural Networks

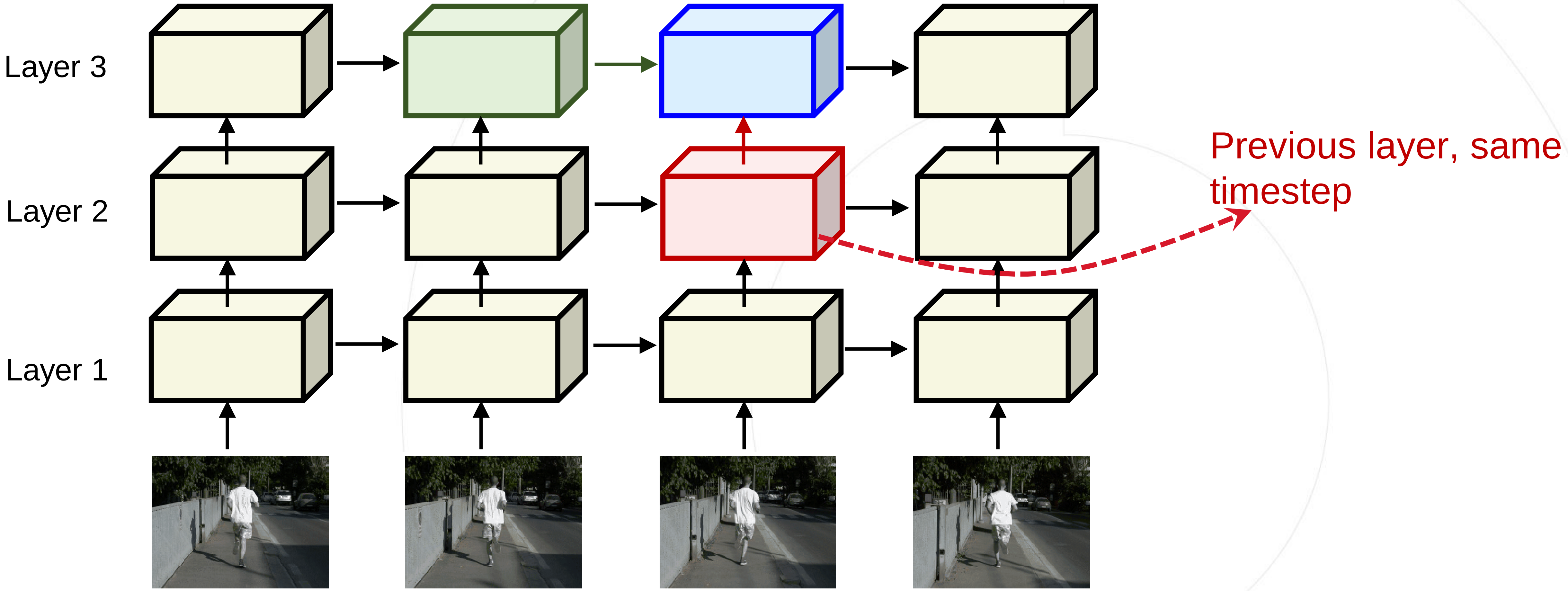


Entire network uses 2D feature maps: C x H x W

Based on Slides of Justin Johnson

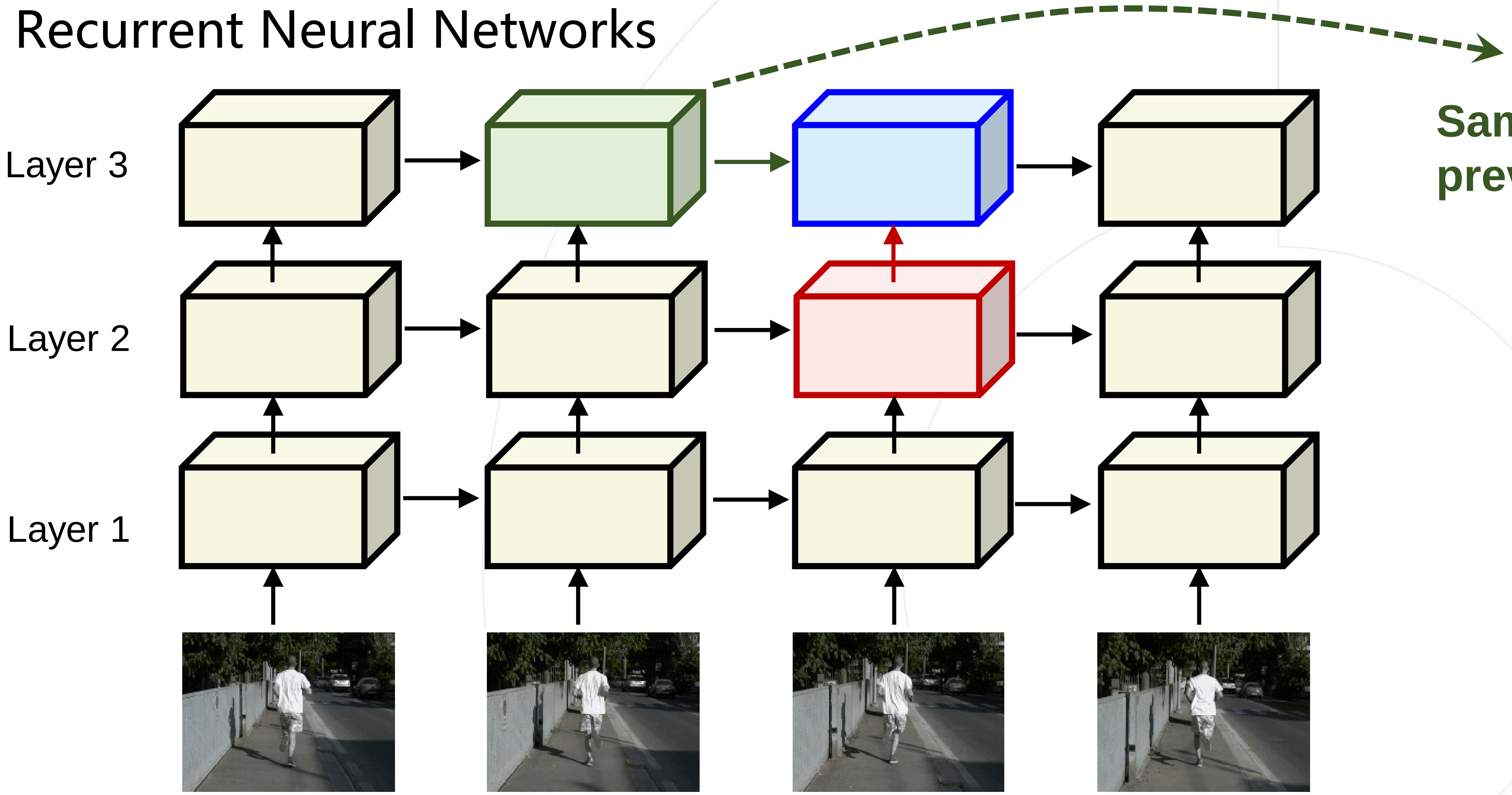
Tsinghua University sensetime

Recurrent Neural Networks



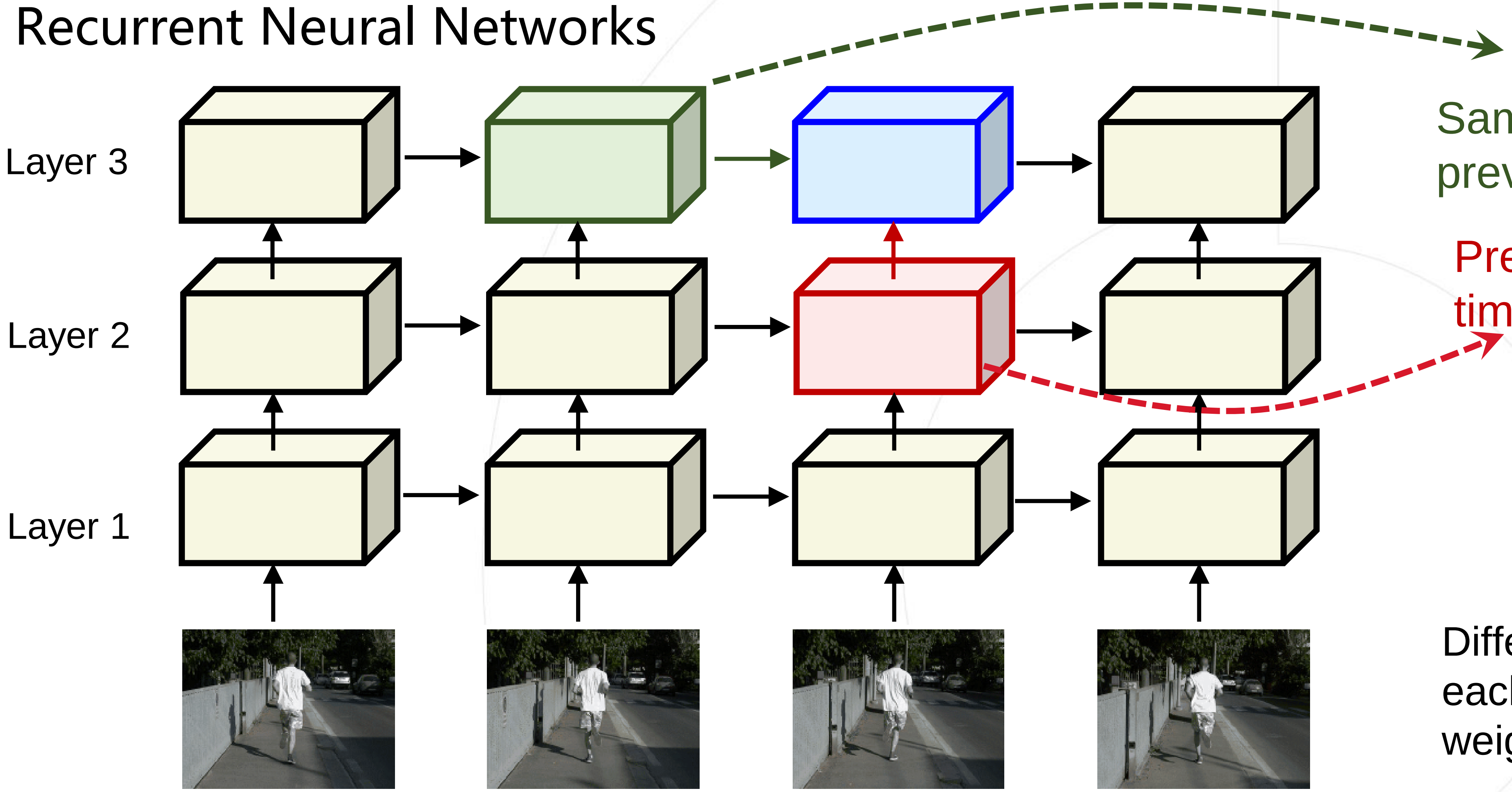
Based on Slides of Justin Johnson

Recurrent Neural Networks



Same layer, previous timestep

Recurrent Neural Networks



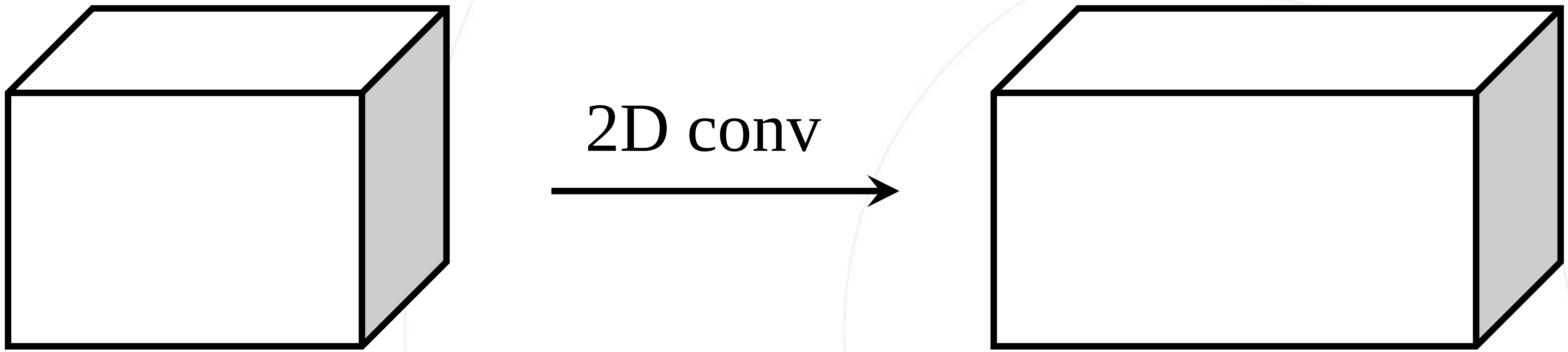
Same layer, previous timestep

Previous layer, same timestep

Different weights at each layer, share weights across time

Based on Slides of Justin Johnson

2D convolution

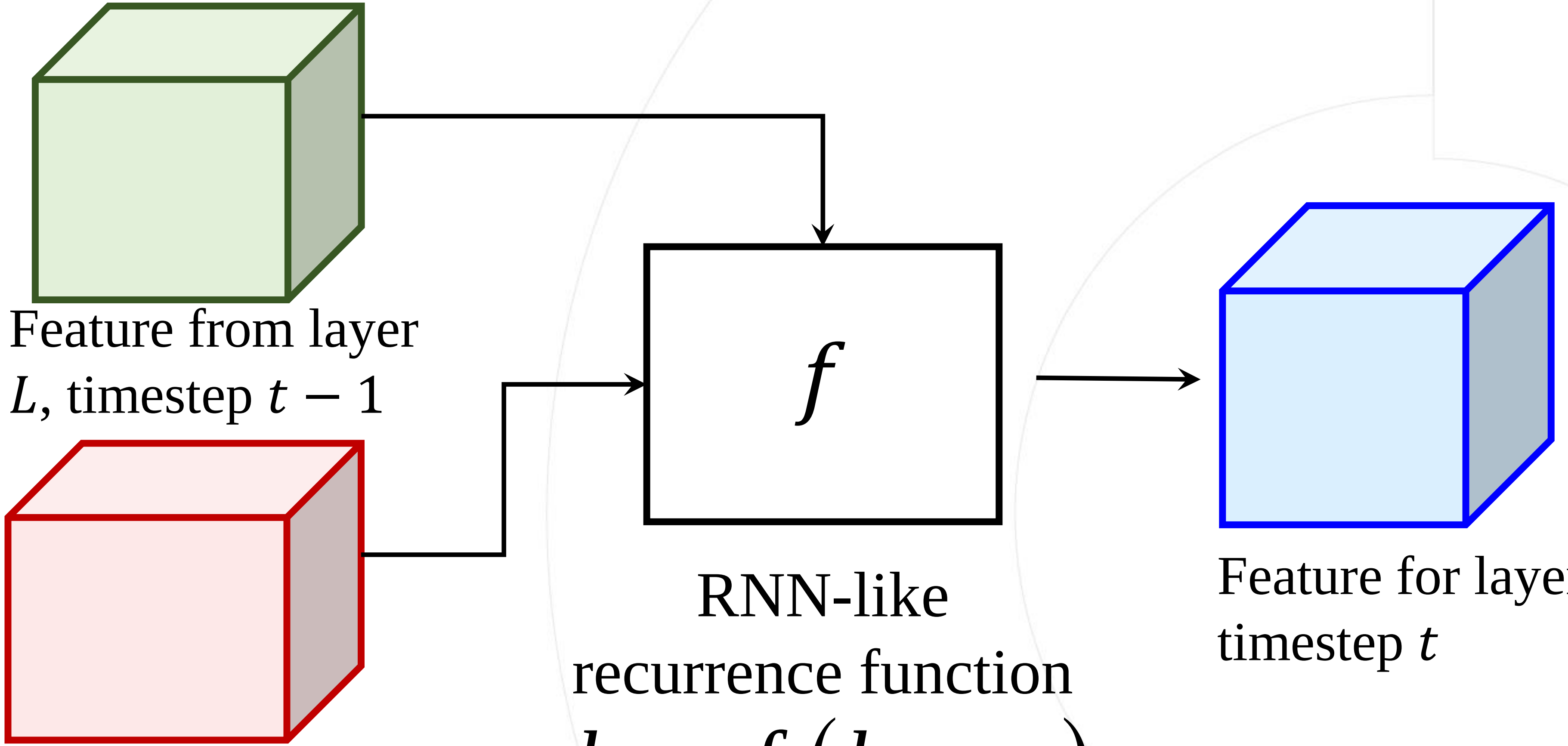


Input features:
$C \times H \times W$

Output features:
$C \times H \times W$

Recurrent Neural Networks

Vanilla RNN: $h_t = \tanh\left(W_h h_{t\text{-}1} + W_x x_t\right)$



Feature from layer L, timestep $t-1$

Feature from layer $L-1$, timestep t

RNN-like recurrence function

$$h_t = f\left(h_{t\text{-}1}, x_t\right)$$

Feature for layer L, timestep t

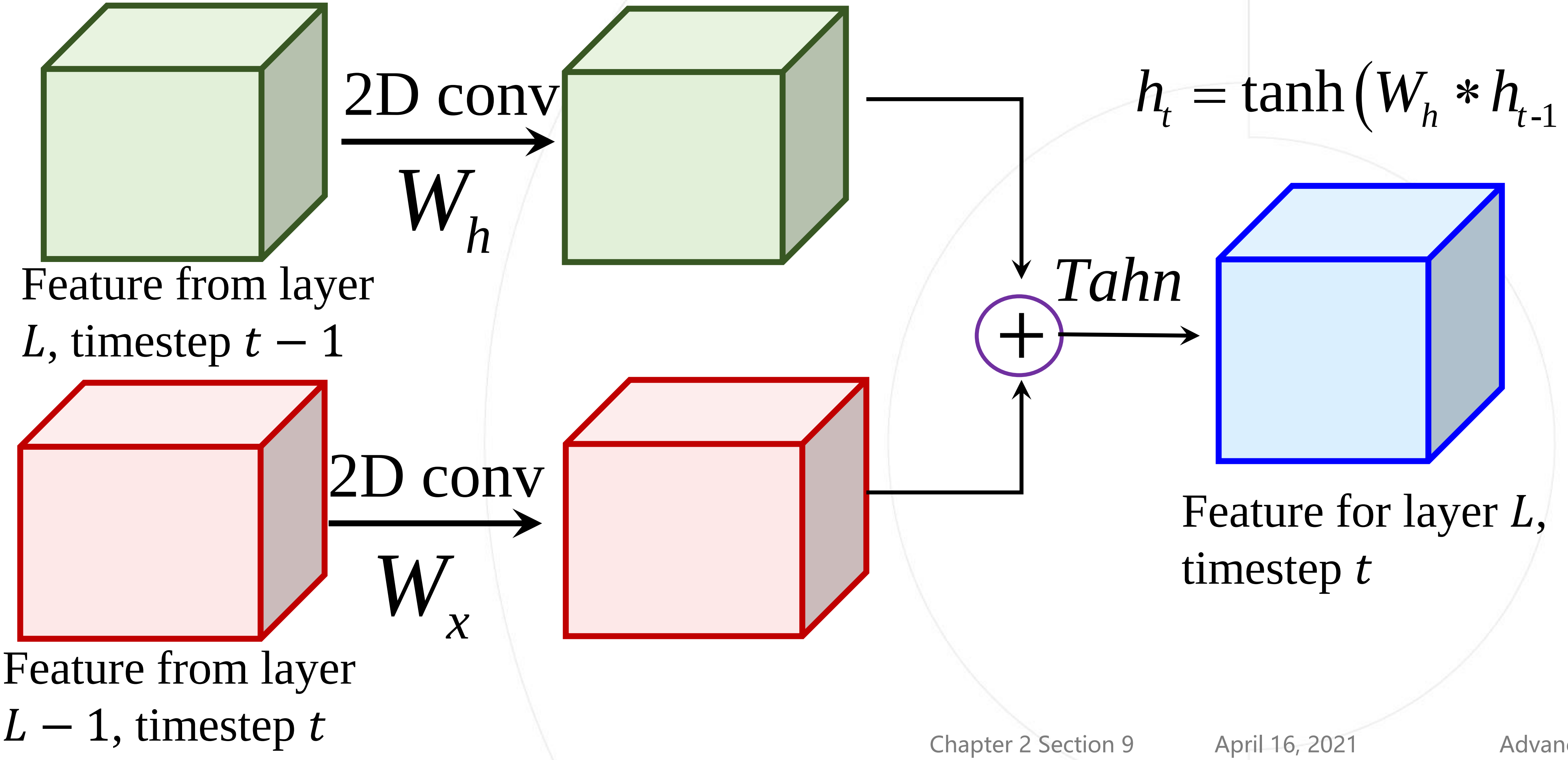
Based on Slides of Justin Johnson

Tsinghua University sensetime

Recurrent Convolutional Networks

Replace all matrix multiply in Vanilla RNN with convolution

$$h_t = \tanh\left(W_h * h_{t\text{-}1} + W_x * x_t\right)$$

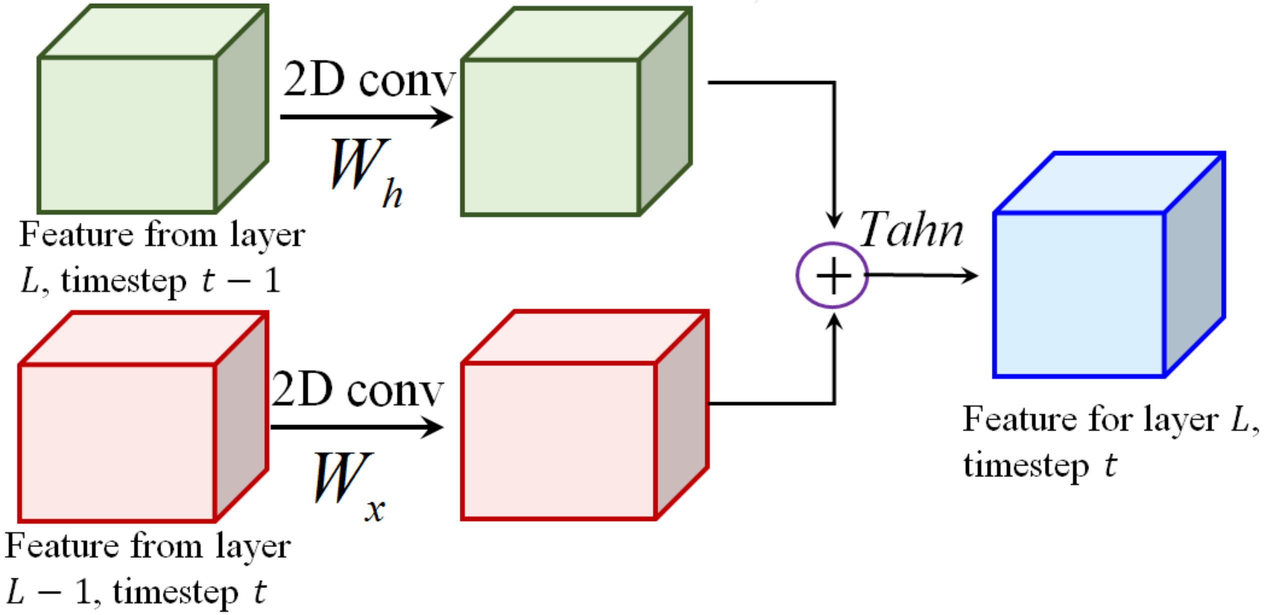


Based on Slides of Justin Johnson

Video classification: CNN+RNN/LSTM

Tsinghua University sensetime

Recurrent Convolutional Networks



GRU

$$r_t = \sigma\left(W_{xr} * x_t + W_{hr} * h_{t-1} + b_r\right)$$

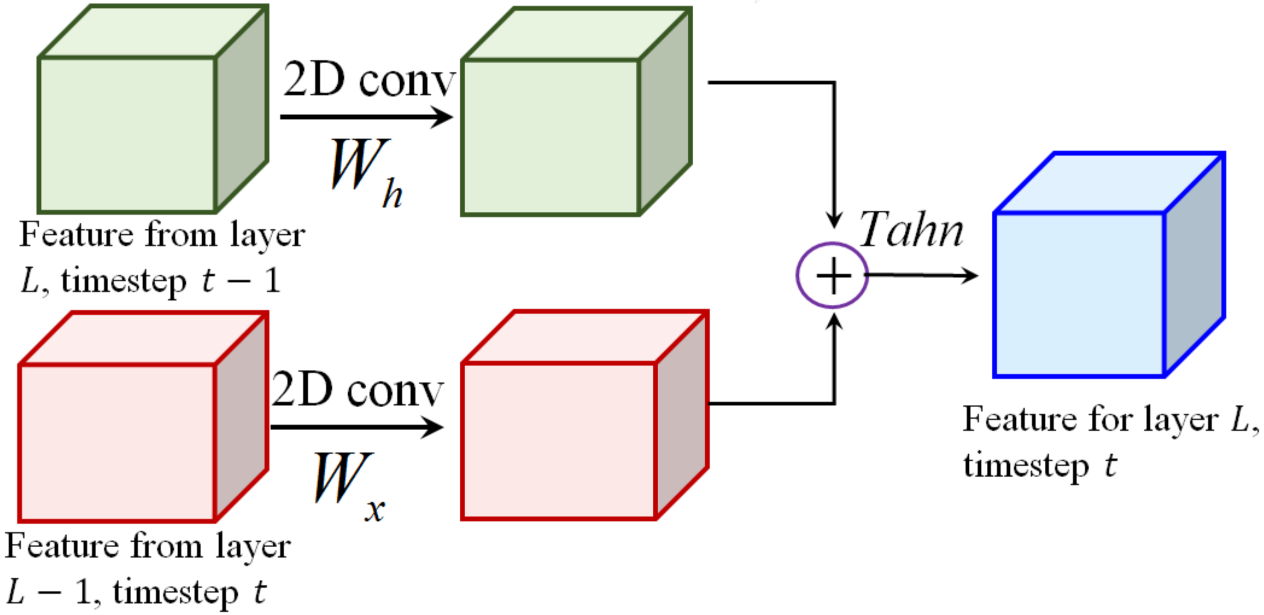
$$z_t = \sigma\left(W_{xz} * x_t + W_{hz} * h_{t-1} + b_z\right)$$

$$\tilde{h}_t = \tanh\left(W_{xh} * x_t + W_{hh} * \left(r_t \odot h_{t-1}\right) + b_h\right)$$

$$h_t = z_t \odot h_{t-1} + \left(1 - z_t\right) \odot \tilde{h}_t$$

Do similar transform for other RNN variants

Recurrent Convolutional Networks



LSTM

$$g_t = \tanh\left(W_{xg} * x_t + W_{hg} * h_{t-1} + b_g\right)$$

$$i_t = \sigma\left(W_{xi} * x_t + W_{hi} * h_{t-1} + b_i\right)$$

$$f_t = \sigma\left(W_{xf} * x_t + W_{hf} * h_{t-1} + b_f\right)$$

$$o_t = \sigma\left(W_{xo} * x_t + W_{ho} * h_{t-1} + b_o\right)$$

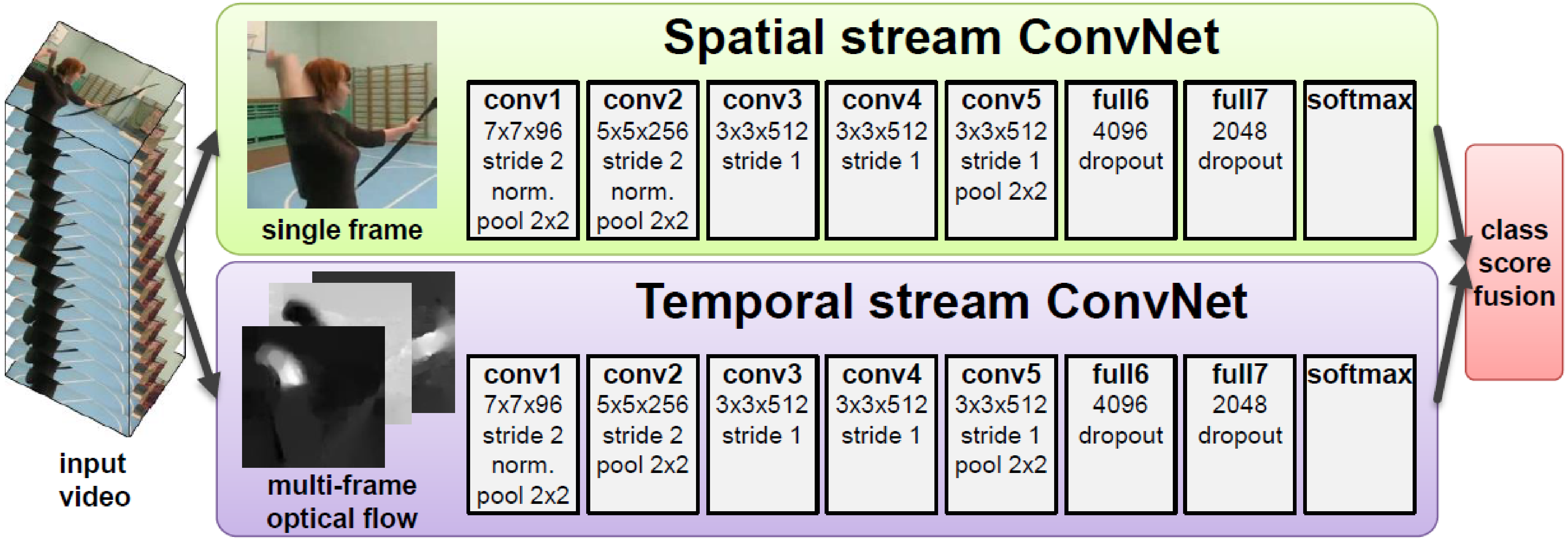
$$c_t = i_t \odot g_t + f_t \odot c_{t-1}$$

$$h_t = o_t \odot \tanh\left(c_t\right)$$

Do similar transform for other RNN variants

Two-stream networks - separating motion & appearance

- **Spatial stream:** performs action recognition from still video frames
- **Temporal stream:** recognizes action from motion in the form of dense optical flow.

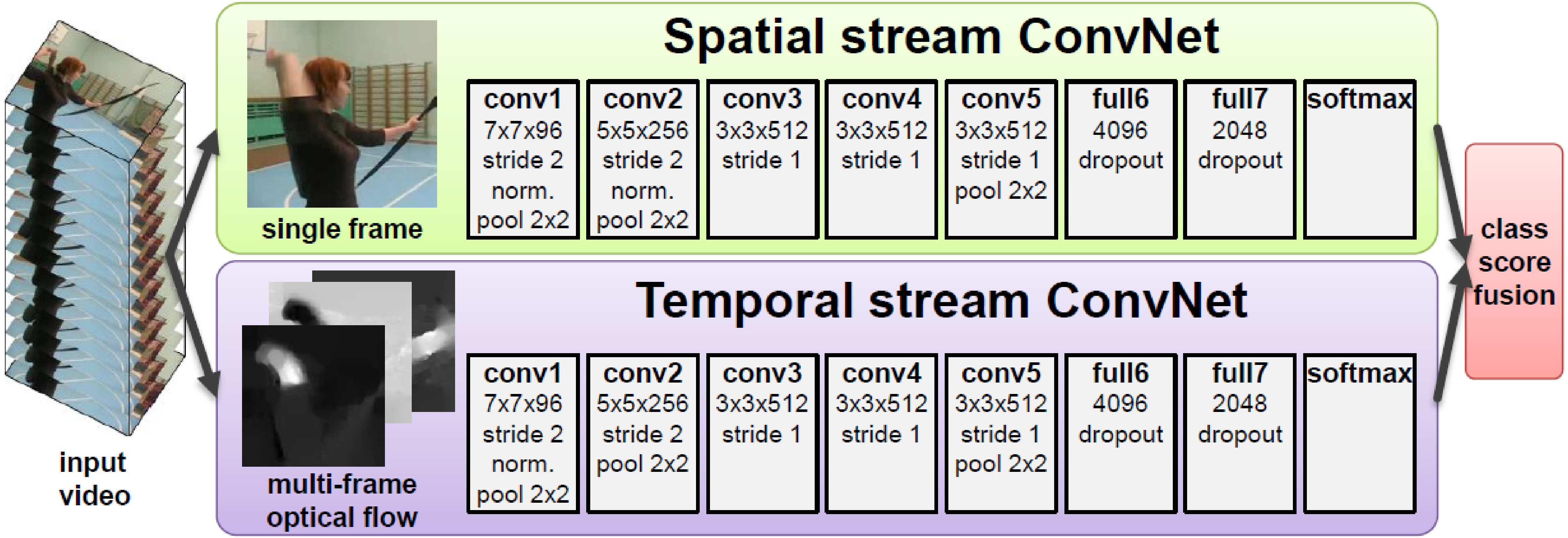


Karen Simonyan and Andrew Zisserman. Two-stream convolutional networks for action recognition in videos. In NIPS, 2014.

Video classification: Two-stream Networks

Video classification: two-stream networks – separating motion & appearance

- **Input of spatial stream:** single image $3 \times H \times w$
- **Input of temporal stream:** stack of optical flow $2(T-1) \times H \times w$, first 2D conv processes all flow images



Karen Simonyan and Andrew Zisserman. Two-stream convolutional networks for action recognition in videos. In NIPS, 2014.

Tsinghua University sensetime

Optical Flow: measuring motion

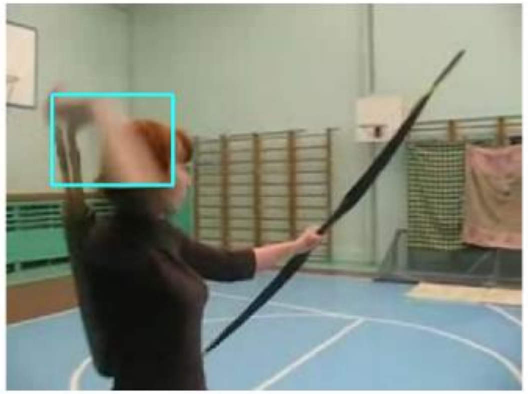


Image frame at time t

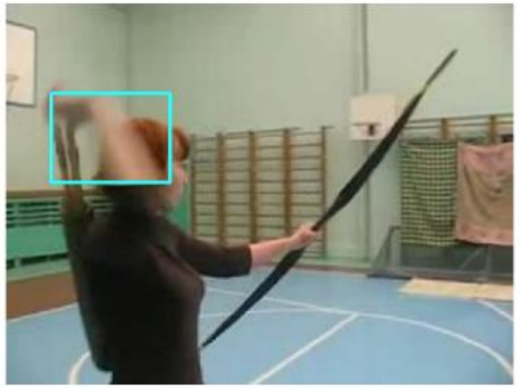
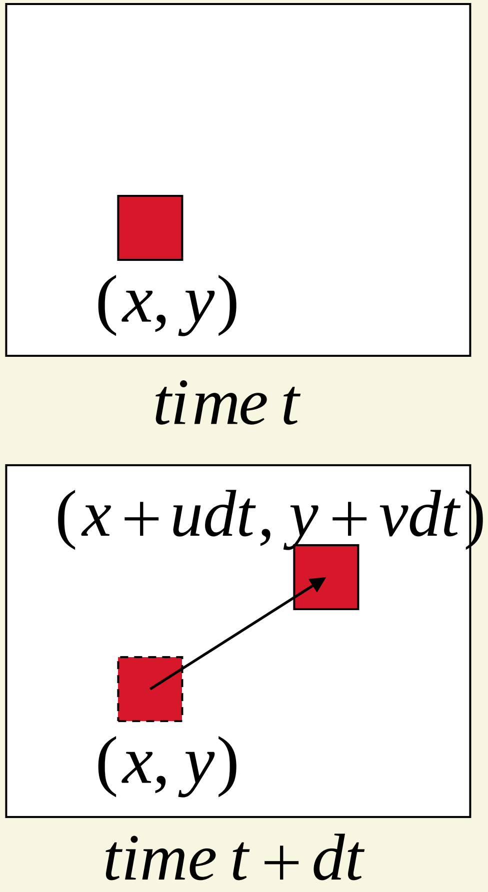


Image frame at time $t+1$



Optical Flow: Velocities (u, v)

Displacement: $(dx, dy) = (udt, vdt)$

- **Brightness constancy:** assume brightness of patch remains same in both images:

$$I(x+dx, y+dy, t+dt) = I(x, y, t)$$

- **Small motion:** assume brightness of patch remains same in both images:

$$I(x+dx, y+dy, t+dt)$$

$$\approx I(x, y, t) + \frac{\partial I}{\partial x} dx + \frac{\partial I}{\partial y} dy + \frac{\partial I}{\partial t} dt$$

Video classification: Two-stream Networks

Optical Flow: measuring motion

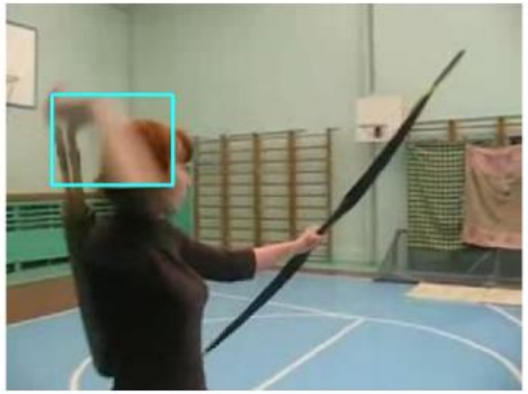


Image frame at time t

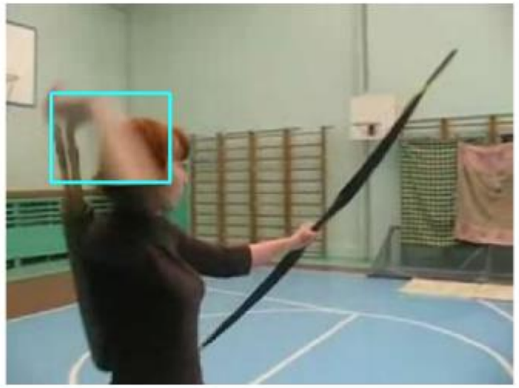
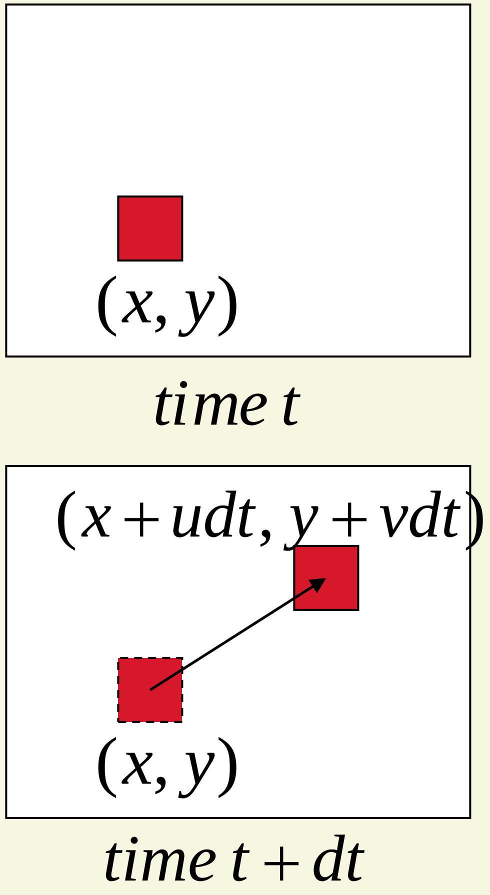


Image frame at time $t+1$



Optical Flow: Velocities (u, v)

Displacement: $(dx, dy) = (udt, vdt)$

- **Brightness constancy:** assume brightness of patch remains same in both images

$$I(x+dx, y+dy, t+dt) = I(x, y, t)$$

- **Small motion:** assume brightness of patch remains same in both images

$$\cancel{I(x,y,t)} + \frac{\partial I}{\partial x}dx + \frac{\partial I}{\partial y}dy + \frac{\partial I}{\partial t}dt = \cancel{I(x,y,t)}$$

Optical Flow: measuring motion

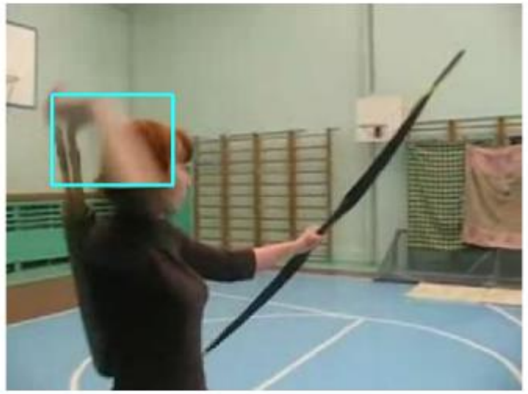


Image frame at time t

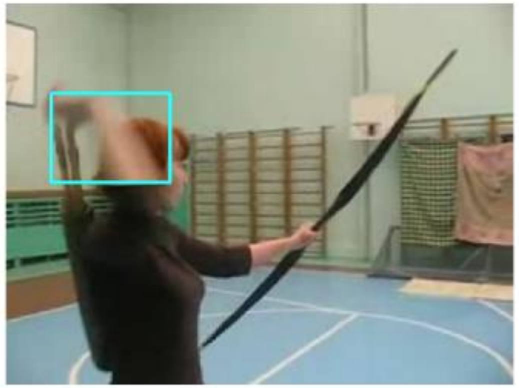
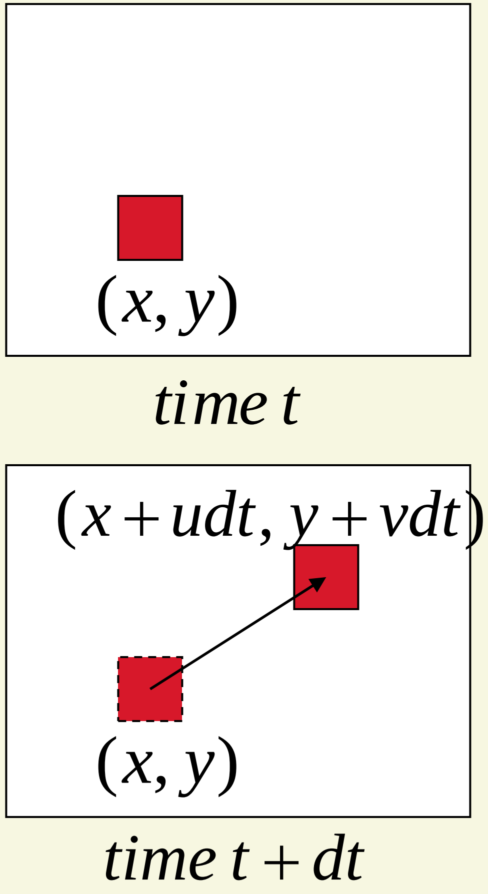


Image frame at time $t+1$



Optical Flow: Velocities (u, v)

Displacement: $(dx, dy) = (udt, vdt)$

- **Brightness constancy:** assume brightness of patch remains same in both images

$$I(x+dx, y+dy, t+dt) = I(x, y, t)$$

- **Small motion:** assume brightness of patch remains same in both images

$$I_x dx + I_y dy + I_t dt = 0 \Rightarrow I_x \frac{dx}{dt} + I_y \frac{dy}{dt} + I_t = 0$$

$$\Rightarrow I_x u + I_y v + I_t = 0$$

Tsinghua University sensetime

Optical Flow: measuring motion

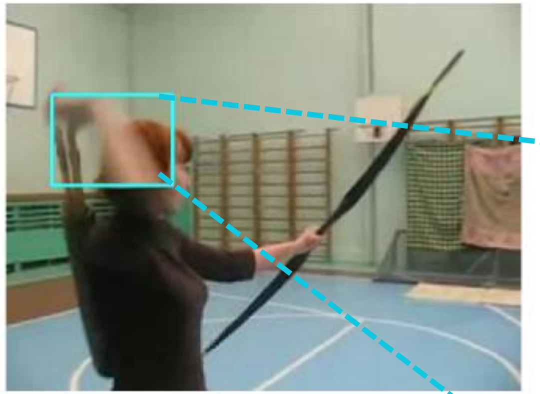


Image frame at time t

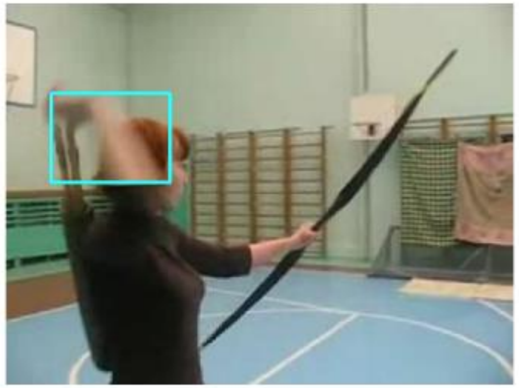
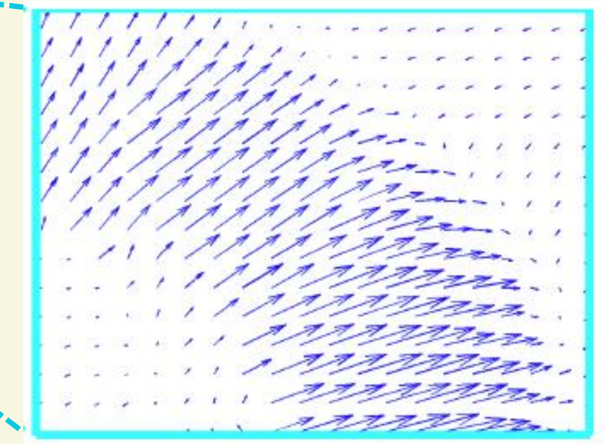


Image frame at time $t + 1$

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



Optical flow tells where each pixel will move in the next frames

$$F(x,y) = (u,v)$$

$$I_{t+1}(x+u, y+v) = I_t(x,y)$$

Based on Slides of Justin Johnson

Optical Flow: measuring motion

Optical flow highlights local motion.

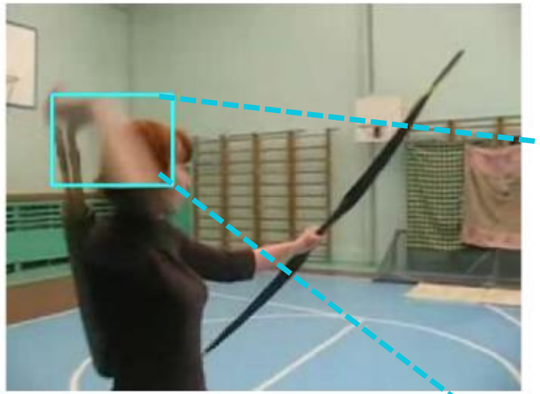


Image frame at time t

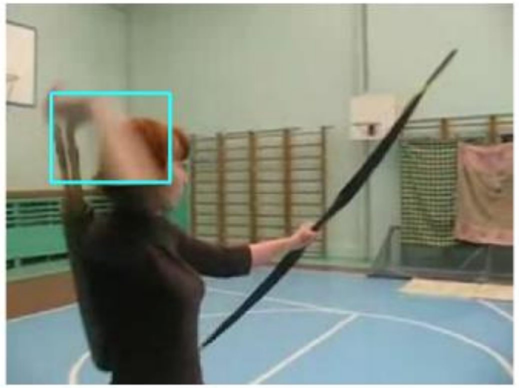
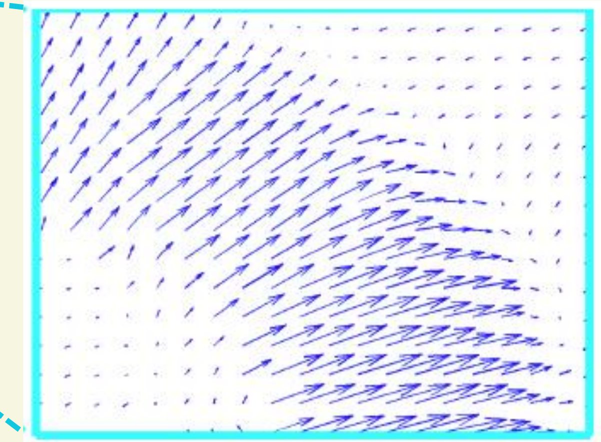


Image frame at time $t+1$

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



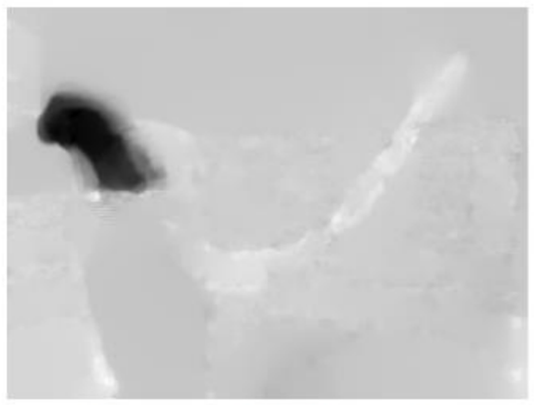
Optical flow tells where each pixel will move in the next frames

$$F(x,y)=(u,v)$$

$$I_{t+1}(x+u,y+v)=I_t(x,y)$$

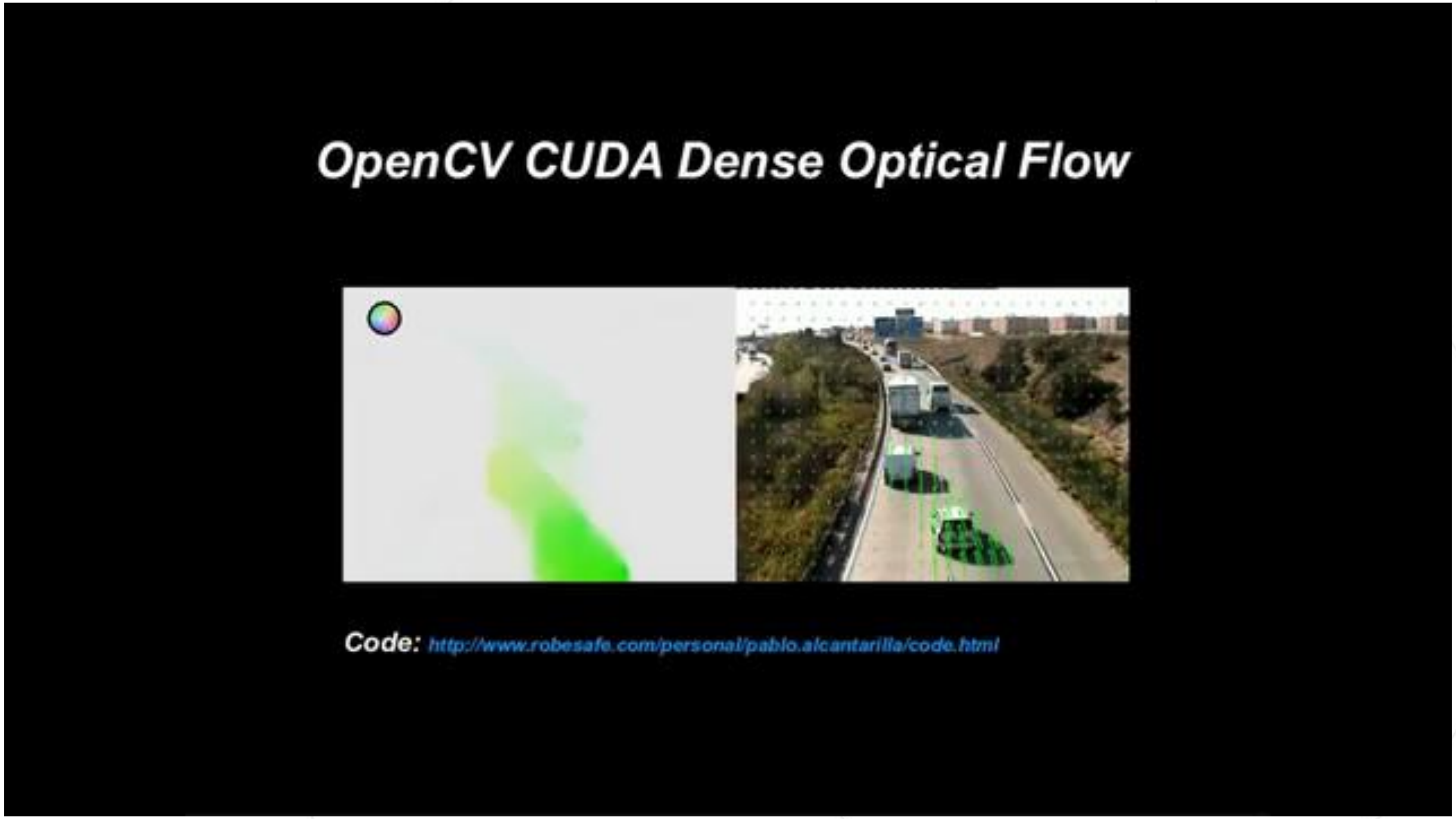
Horizontal flow u

Vertical flow v

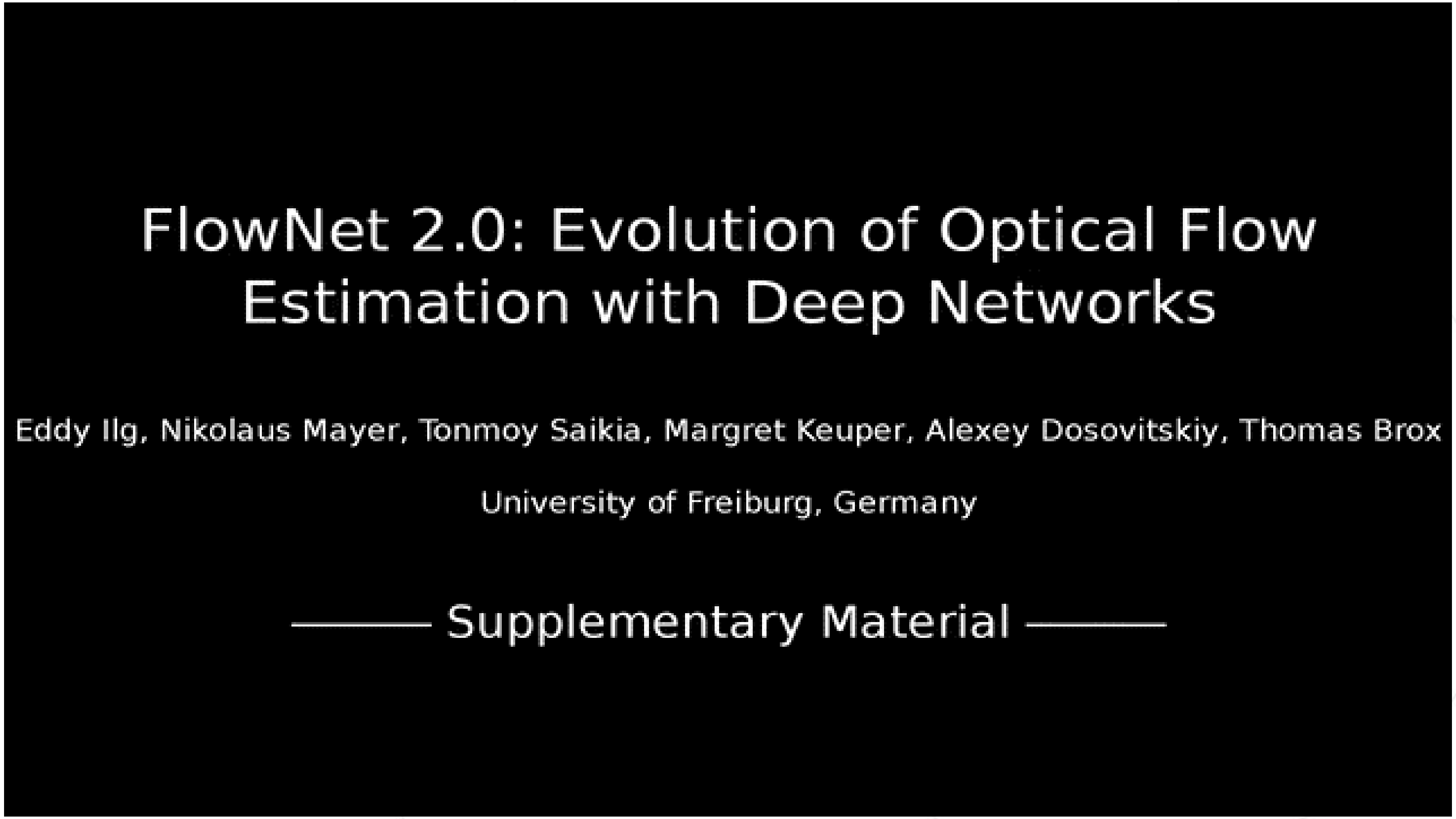


Based on Slides of Justin Johnson

Optical Flow: measuring motion



Optical Flow: measuring motion



Optical Flow: measuring motion

a) Optical flow stacking; b) Trajectory stacking;
c) Bi-directional optical flow; d) Mean flow subtraction.

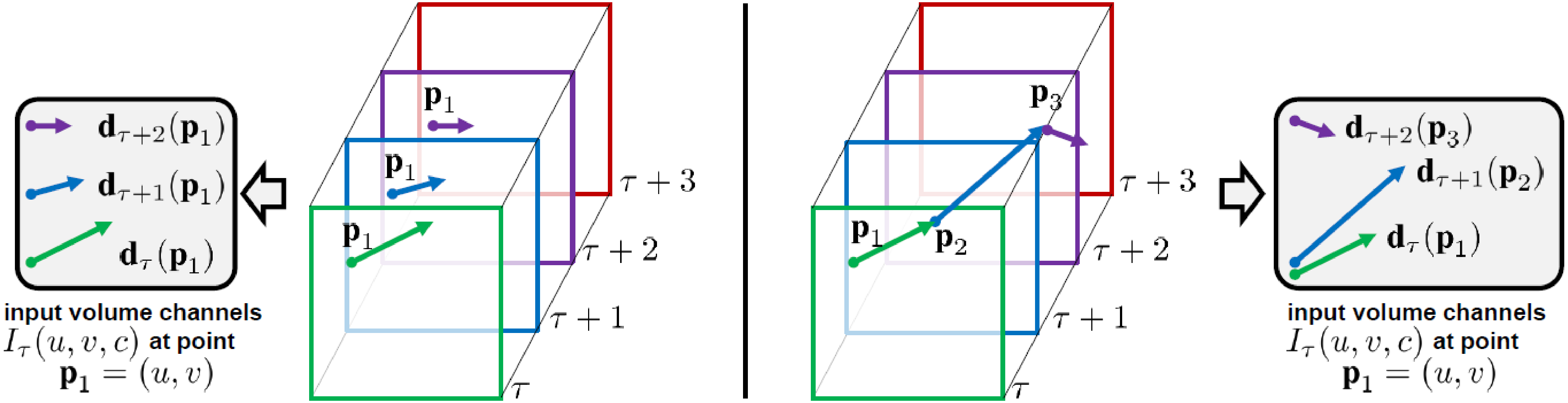
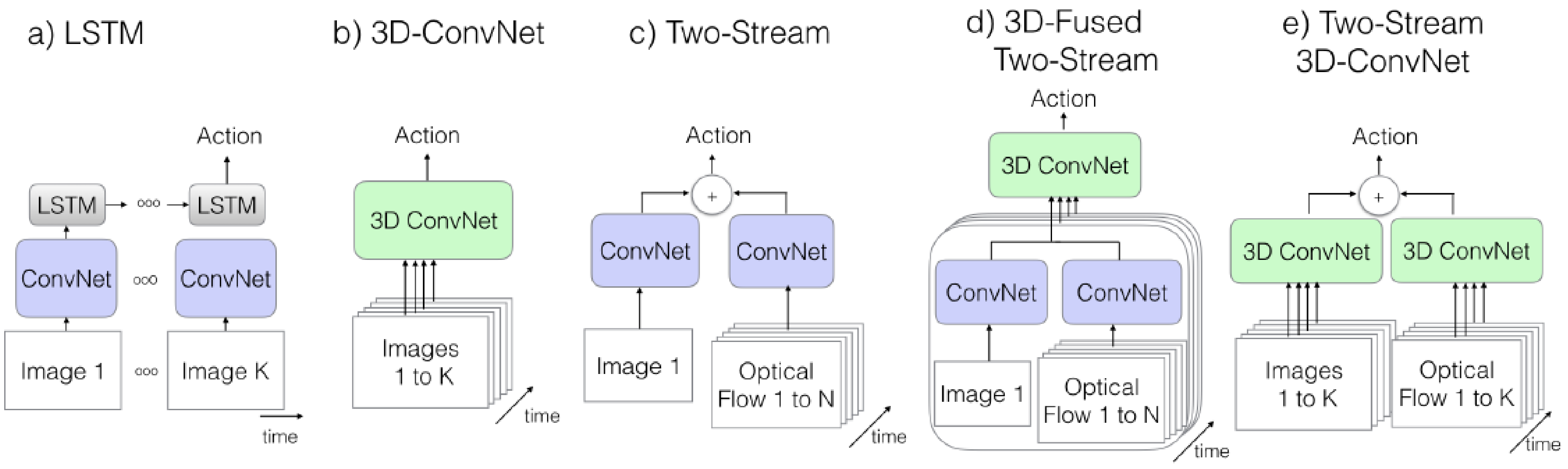


Figure 3: **ConvNet input derivation from the multi-frame optical flow.** *Left:* optical flow stacking (1) samples the displacement vectors **d** at the same location in multiple frames. *Right:* trajectory stacking (2) samples the vectors along the trajectory. The frames and the corresponding displacement vectors are shown with the same colour.

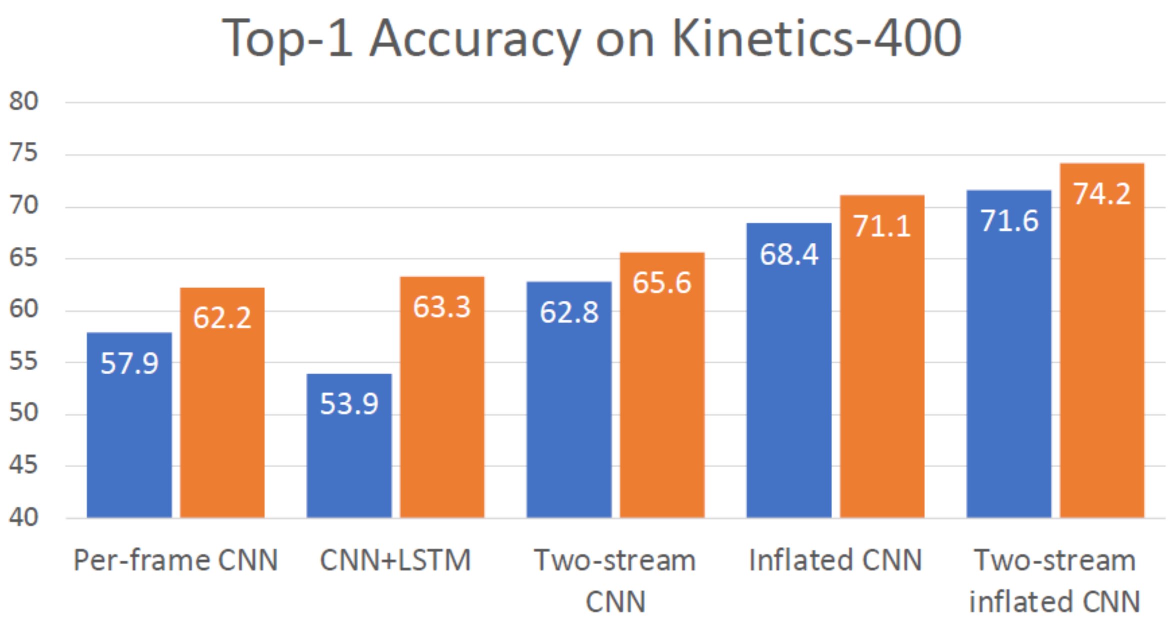
Karen Simonyan and Andrew Zisserman. Two-stream convolutional networks for action recognition in videos. In NIPS, 2014.

Video classification: I3D (Two-stream)



Carreira et al.. Quo Vadis, Action Recognition? A New Model and the Kinetics Dataset. In Proc. CVPR , 2017.

Video classification: performance comparison

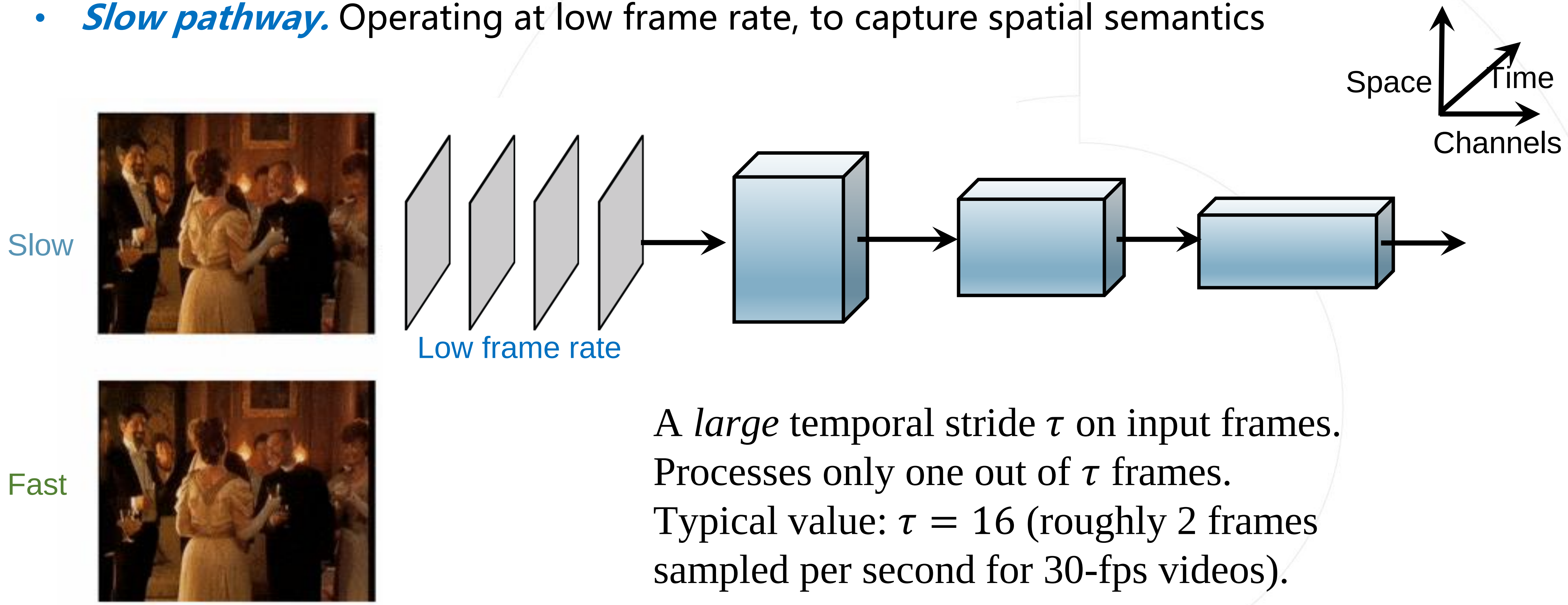


method	pretraining dataset	top1	top5
I3D-RGB [4]	none	67.5	87.2
I3D-RGB [4]	ImageNet	72.1	90.3
I3D-Flow [4]	ImageNet	65.3	86.2
I3D-Two-Stream [4]	ImageNet	**75.7**	**92.0**
R(2+1)D-RGB	none	72.0	90.0
R(2+1)D-Flow	none	67.5	87.2
R(2+1)D-Two-Stream	none	73.9	90.9
R(2+1)D-RGB	Sports-1M	74.3	91.4
R(2+1)D-Flow	Sports-1M	68.5	88.1
R(2+1)D-Two-Stream	Sports-1M	**75.4**	**91.9**

Table 5. **Comparison with the state-of-the-art on Kinetics.** R(2+1)D outperforms I3D by 4.5% when trained from scratch on RGB. R(2+1)D pretrained on Sports-1M outperforms I3D pre-trained on ImageNet, for both RGB and optical flow. However, it is slightly worse than I3D (0.3%) when fusing the two streams.

SlowFast Networks: treating time and space differently

- ***Slow pathway.*** Operating at low frame rate, to capture spatial semantics

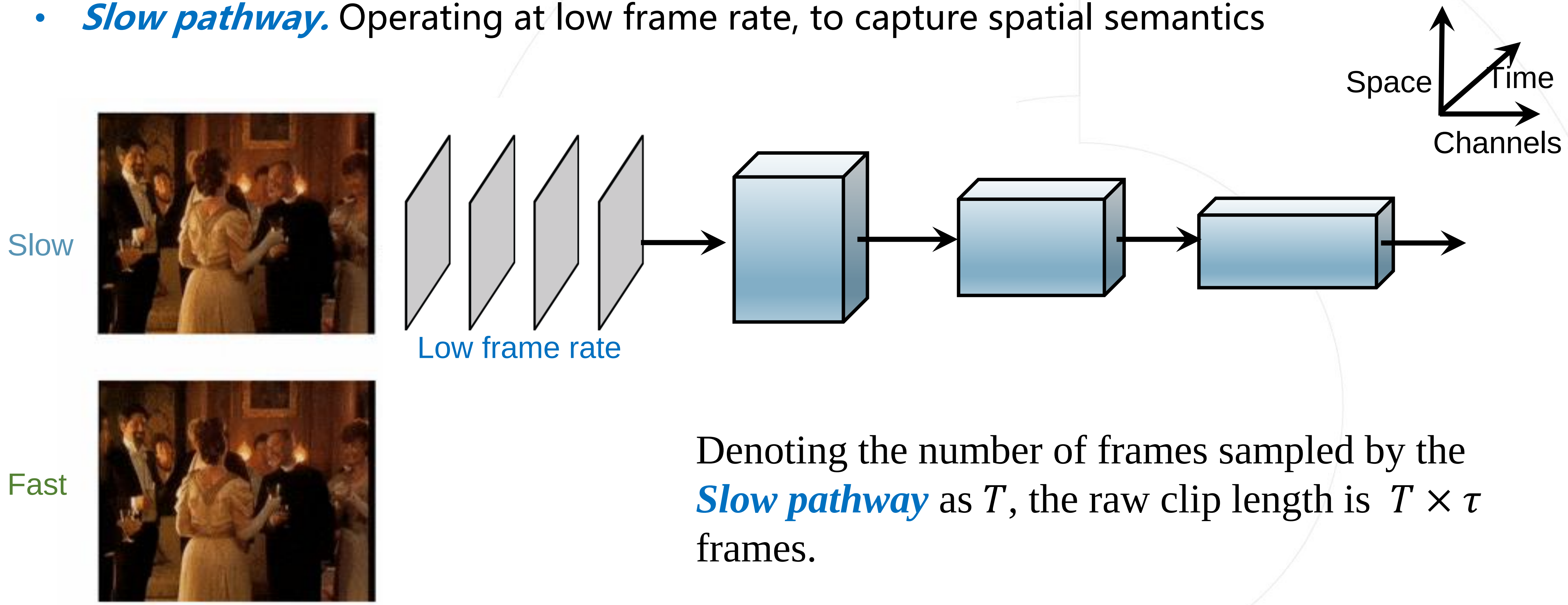


A *large* temporal stride τ on input frames.
Processes only one out of τ frames.
Typical value: $\tau = 16$ (roughly 2 frames sampled per second for 30-fps videos).

C. Feichtenhofer. SlowFast Networks for Video Recognition. In ICCV 2019.

SlowFast Networks: treating time and space differently

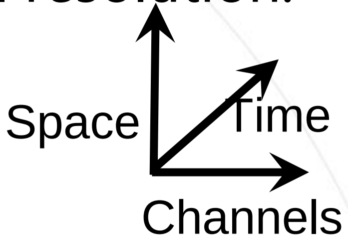
- ***Slow pathway.*** Operating at low frame rate, to capture spatial semantics



Denoting the number of frames sampled by the ***Slow pathway*** as T, the raw clip length is $T \times \tau$ frames.

SlowFast Networks: treating time and space differently

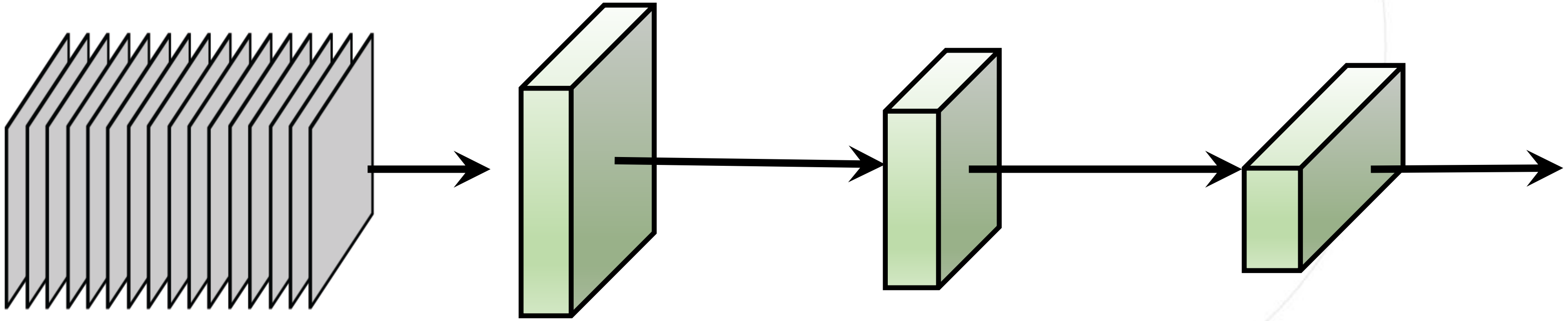
- ***Fast pathway.*** Operates at high frame rate, to capture motion at fine temporal resolution.



High frame rate. Works with a *small* temporal stride of τ/α, where $\alpha > 1$ is the frame rate ratio between the Fast and Slow pathways.

Slow

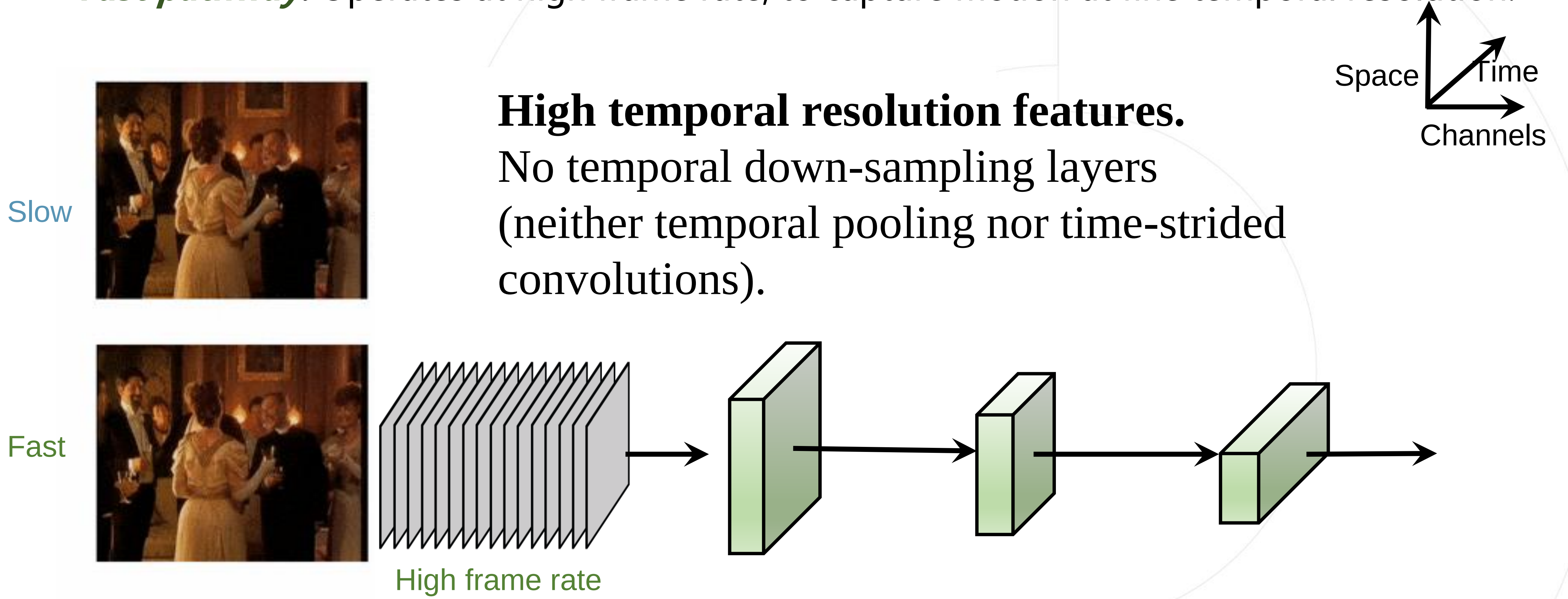
Fast



High frame rate

SlowFast Networks: treating time and space differently

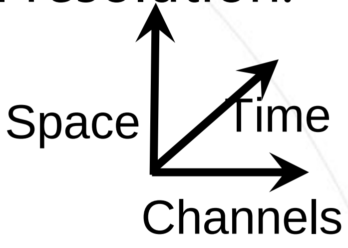
- ***Fast pathway.*** Operates at high frame rate, to capture motion at fine temporal resolution.



High temporal resolution features.
No temporal down-sampling layers (neither temporal pooling nor time-strided convolutions).

SlowFast Networks: treating time and space differently

- ***Fast pathway.*** Operates at high frame rate, to capture motion at fine temporal resolution.



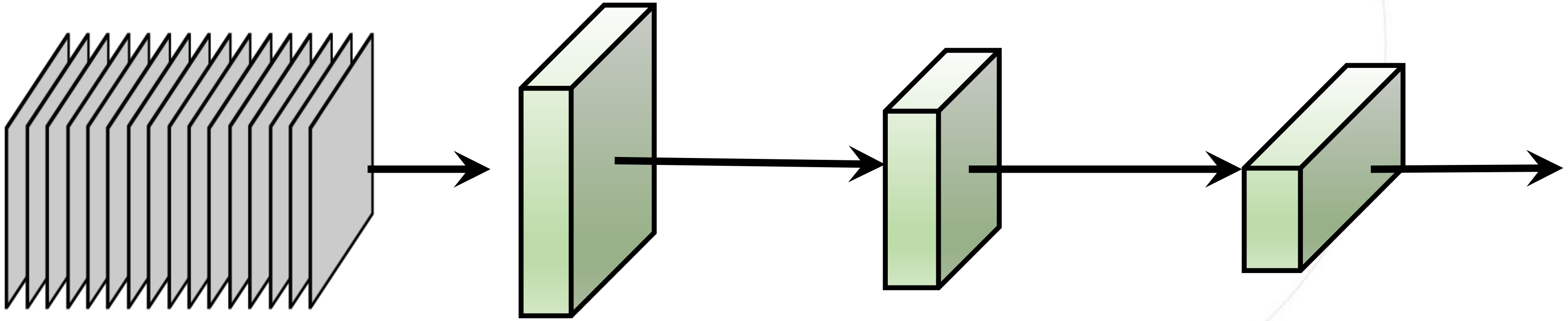
Low channel capacity. Analogous to the Slow pathway, but has a ratio of β ($\beta < 1$) channels of the Slow pathway.

Typical value: $\beta = 1/8$.

~20% of the total computation

Slow

Fast

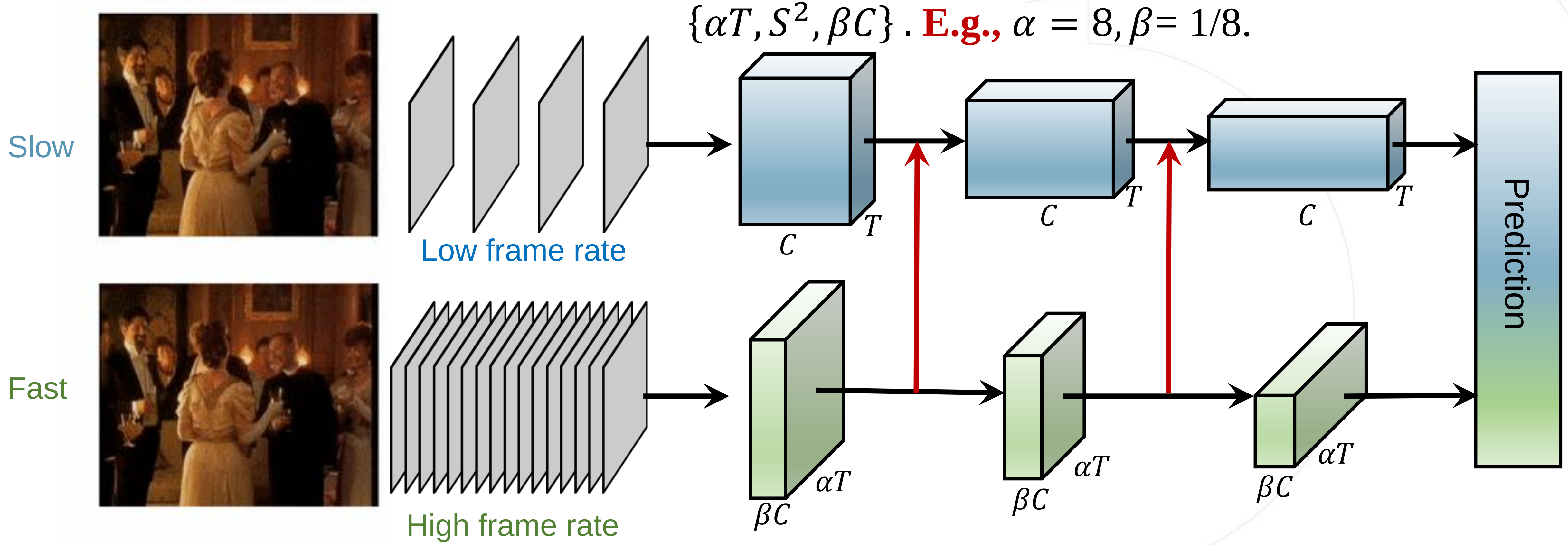


High frame rate

C. Feichtenhofer. SlowFast Networks for Video Recognition. In ICCV 2019.

SlowFast Networks: treating time and space differently

Denoting the feature shape of *Slow pathway* as $\{T, S^2, C\}$, the feature shape of the *Fast pathway* is $\{\alpha T, S^2, \beta C\}$. **E.g.,** $\alpha = 8, \beta = 1/8$.



C. Feichtenhofer. SlowFast Networks for Video Recognition. In ICCV 2019.

Tsinghua University sensetime

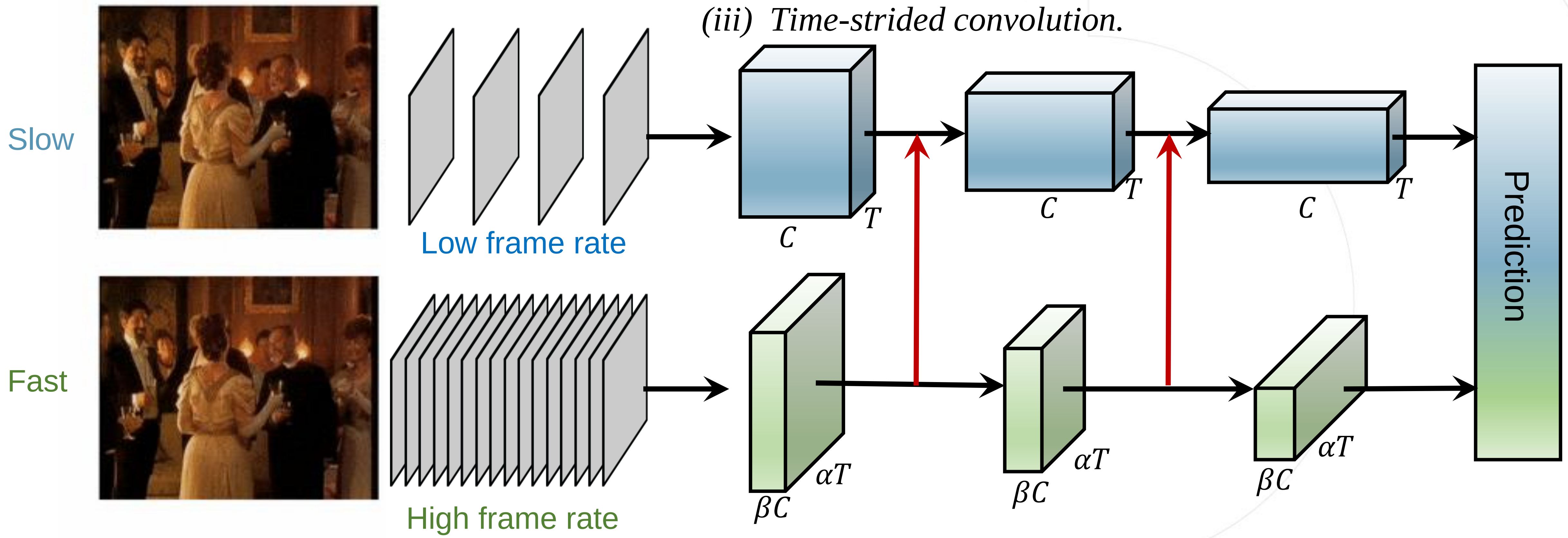
SlowFast Networks: treating time and space differently

Lateral connections.

(i) Time-to-channel: Reshape and transpose.

(ii) Time-strided sampling.

(iii) Time-strided convolution.



C. Feichtenhofer. SlowFast Networks for Video Recognition. In ICCV 2019.

Video classification: SlowFast

SlowFast Networks

- Dimensions are $\{T \times S^2, C\}$
- Strides are {temporal, spatial}
- Residual blocks are shown by brackets
- Non-degenerate temporal filters are underlined
- Orange numbers mark fewer channels, for the Fast pathway
- Green numbers mark higher temporal resolution of the Fast pathway
- No temporal *pooling* is performed throughout the hierarchy

stage	*Slow* pathway	*Fast* pathway	output sizes $T\times S^2$
raw clip	-	-	64×224^2
data layer	stride 16, 1^2	stride 2, 1^2	*Slow* : 4×224^2 *Fast* : 32×224^2
conv_1	1×7^2, 64 stride 1, 2^2	$\underline{5\times7^2}$, 8 stride 1, 2^2	*Slow* : 4×112^2 *Fast* : 32×112^2
pool_1	1×3^2 max stride 1, 2^2	1×3^2 max stride 1, 2^2	*Slow* : 4×56^2 *Fast* : 32×56^2
res_2	$\begin{bmatrix} 1\times1^2, 64 \\ 1\times3^2, 64 \\ 1\times1^2, 256 \end{bmatrix}\times3$	$\begin{bmatrix} \underline{3\times1^2}, 8 \\ 1\times3^2, 8 \\ 1\times1^2, 32 \end{bmatrix}\times3$	*Slow* : 4×56^2 *Fast* : 32×56^2
res_3	$\begin{bmatrix} 1\times1^2, 128 \\ 1\times3^2, 128 \\ 1\times1^2, 512 \end{bmatrix}\times4$	$\begin{bmatrix} \underline{3\times1^2}, 16 \\ 1\times3^2, 16 \\ 1\times1^2, 64 \end{bmatrix}\times4$	*Slow* : 4×28^2 *Fast* : 32×28^2
res_4	$\begin{bmatrix} \underline{3\times1^2}, 256 \\ 1\times3^2, 256 \\ 1\times1^2, 1024 \end{bmatrix}\times6$	$\begin{bmatrix} \underline{3\times1^2}, 32 \\ 1\times3^2, 32 \\ 1\times1^2, 128 \end{bmatrix}\times6$	*Slow* : 4×14^2 *Fast* : 32×14^2
res_5	$\begin{bmatrix} \underline{3\times1^2}, 512 \\ 1\times3^2, 512 \\ 1\times1^2, 2048 \end{bmatrix}\times3$	$\begin{bmatrix} \underline{3\times1^2}, 64 \\ 1\times3^2, 64 \\ 1\times1^2, 256 \end{bmatrix}\times3$	*Slow* : 4×7^2 *Fast* : 32×7^2
global average pool, concate, fc			# classes

C. Feichtenhofer. SlowFast Networks for Video Recognition. In ICCV 2019.

Video classification: SlowFast

Biological studies in the primate visual system.

- **Parvocellular (P-cells):** 80%, provide fine spatial detail and color, but lower temporal resolution, responding slowly to stimuli.
- **Magnocellular (M-cells):** 15-20%, operate at *high temporal frequency*, responsive to fast temporal changes, but not sensitive to spatial detail or color.

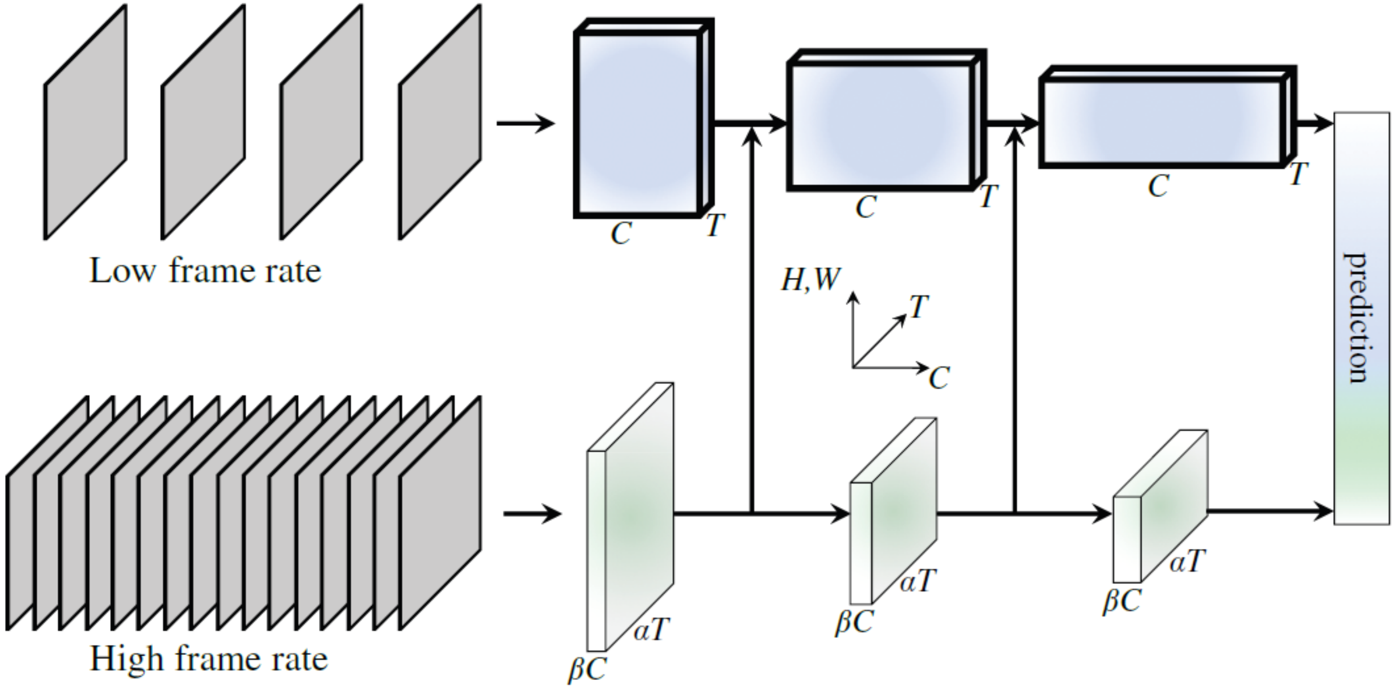


Figure 1. **A SlowFast network** has a low frame rate, low temporal resolution *Slow* pathway and a high frame rate, $\alpha\times$ higher temporal resolution *Fast* pathway. The Fast pathway is lightweight by using a fraction (β, *e.g.*, 1/8) of channels. Lateral connections fuse them.

Video classification: performance comparison

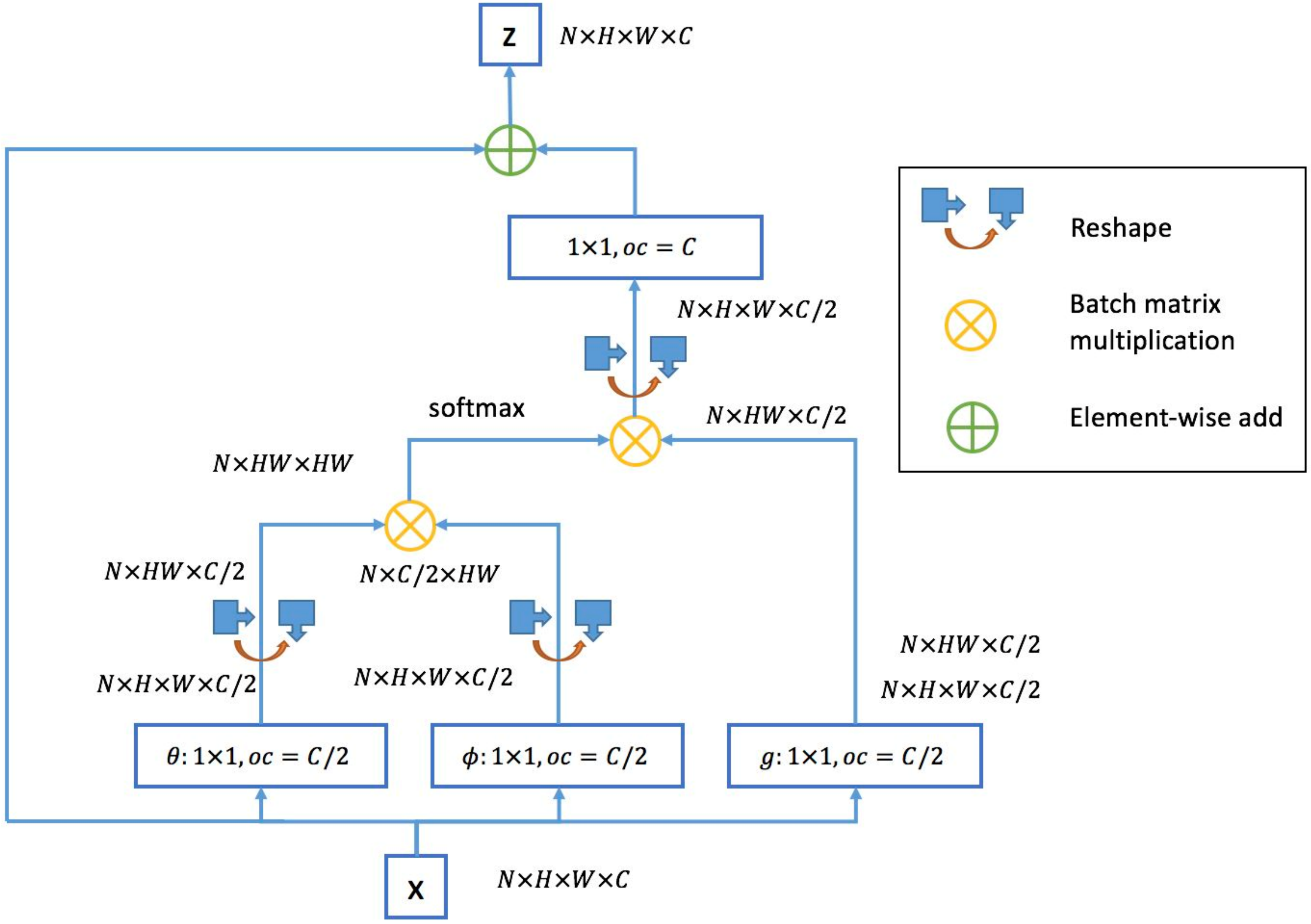
model	flow	pretrain	top-1	top-5	GFLOPs×views
I3D [5]		ImageNet	72.1	90.3	108 × N/A
Two-Stream I3D [5]	✓	ImageNet	75.7	92.0	216 × N/A
S3D-G [61]	✓	ImageNet	77.2	93.0	143 × N/A
Nonlocal R50 [56]		ImageNet	76.5	92.6	282 × 30
Nonlocal R101 [56]		ImageNet	77.7	93.3	359 × 30
R(2+1)D Flow [50]	✓	-	67.5	87.2	152 × 115
STC [9]		-	68.7	88.5	N/A × N/A
ARTNet [54]		-	69.2	88.3	23.5 × 250
S3D [61]		-	69.4	89.1	66.4 × N/A
ECO [63]		-	70.0	89.4	N/A × N/A
I3D [5]	✓	-	71.6	90.0	216 × N/A
R(2+1)D [50]		-	72.0	90.0	152 × 115
R(2+1)D [50]	✓	-	73.9	90.9	304 × 115
SlowFast 4×16, R50		-	75.6	92.1	36.1 × 30
SlowFast 8×8, R50		-	77.0	92.6	65.7 × 30
SlowFast 8×8, R101		-	77.9	93.2	106 × 30
SlowFast 16×8, R101		-	78.9	93.5	213 × 30
SlowFast 16×8, R101+NL		-	**79.8**	**93.9**	234 × 30

Comparison on Kinetics-400. In the last column, we report the inference cost with a single “view“ (temporal clip with spatial crop) × the numbers of such views used. The SlowFast models are with different input sampling (T×) and backbones (R-50, R-101, NL). “N/A” indicates the numbers are not available for us.

Spatio-Temporal Self-Attention (Nonlocal Block)

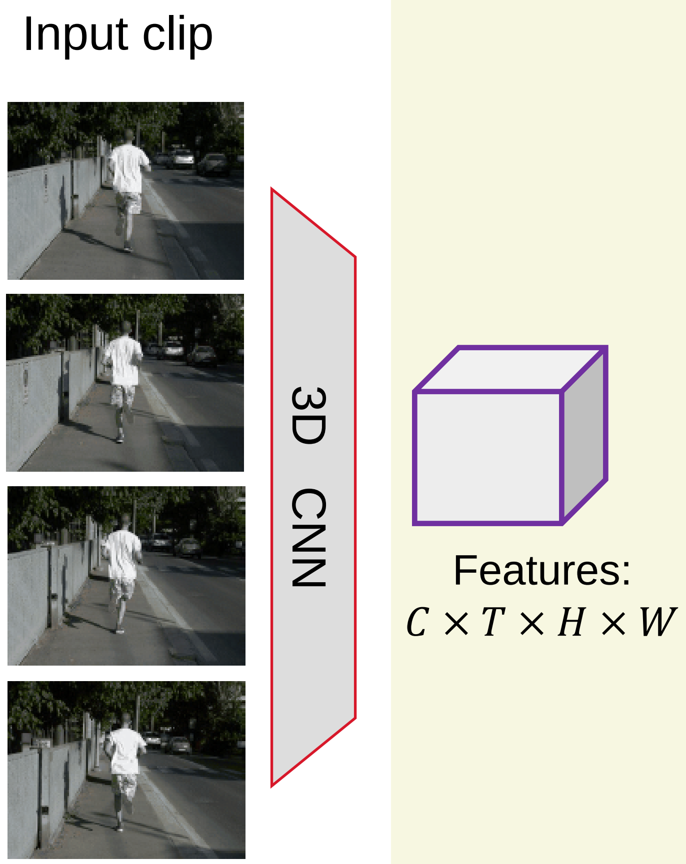
Wang et al, "Non-local neural networks", CVPR 2018

Spatio-Temporal Self-Attention (Nonlocal Block)



Wang et al, "Non-local neural networks", CVPR 2018

Video classification: Spatio-Temporal Self-Attention (Nonlocal Block)

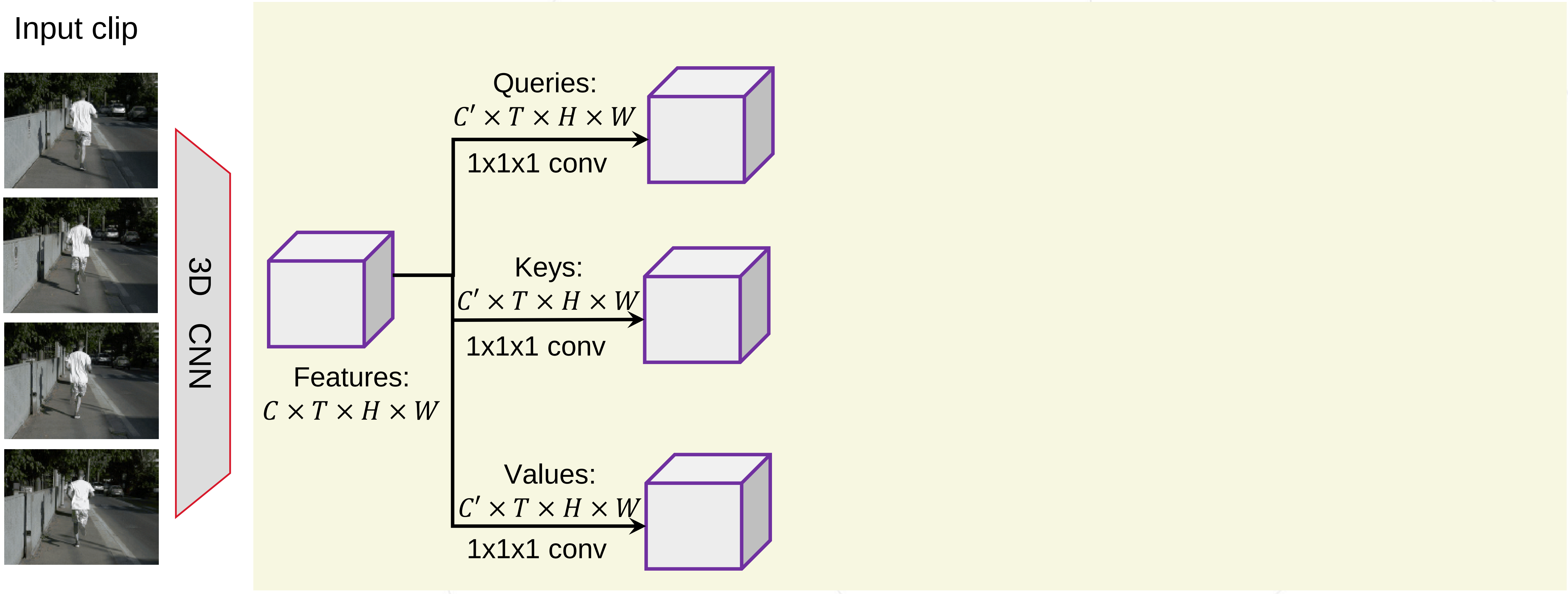


Nonlocal Block

Based on Slides of Justin Johnson

Video classification: Spatio-Temporal Self-Attention (Nonlocal Block)

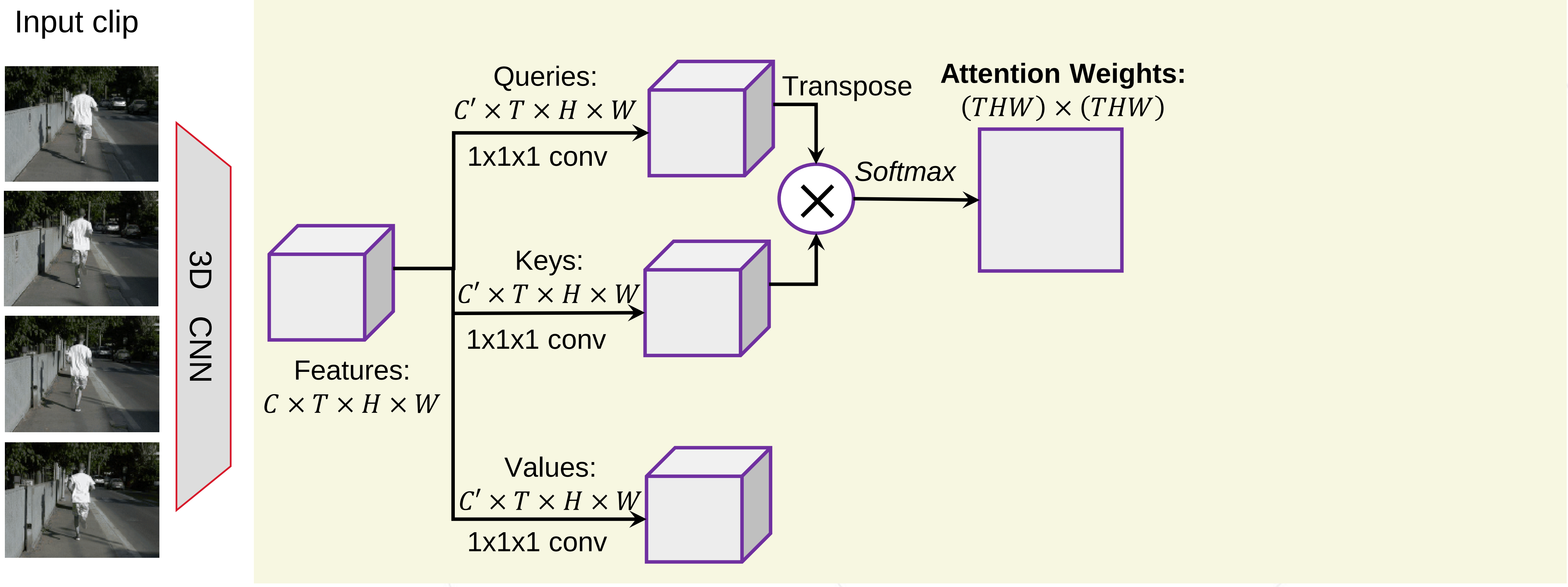
Input clip



Nonlocal Block

Based on Slides of Justin Johnson

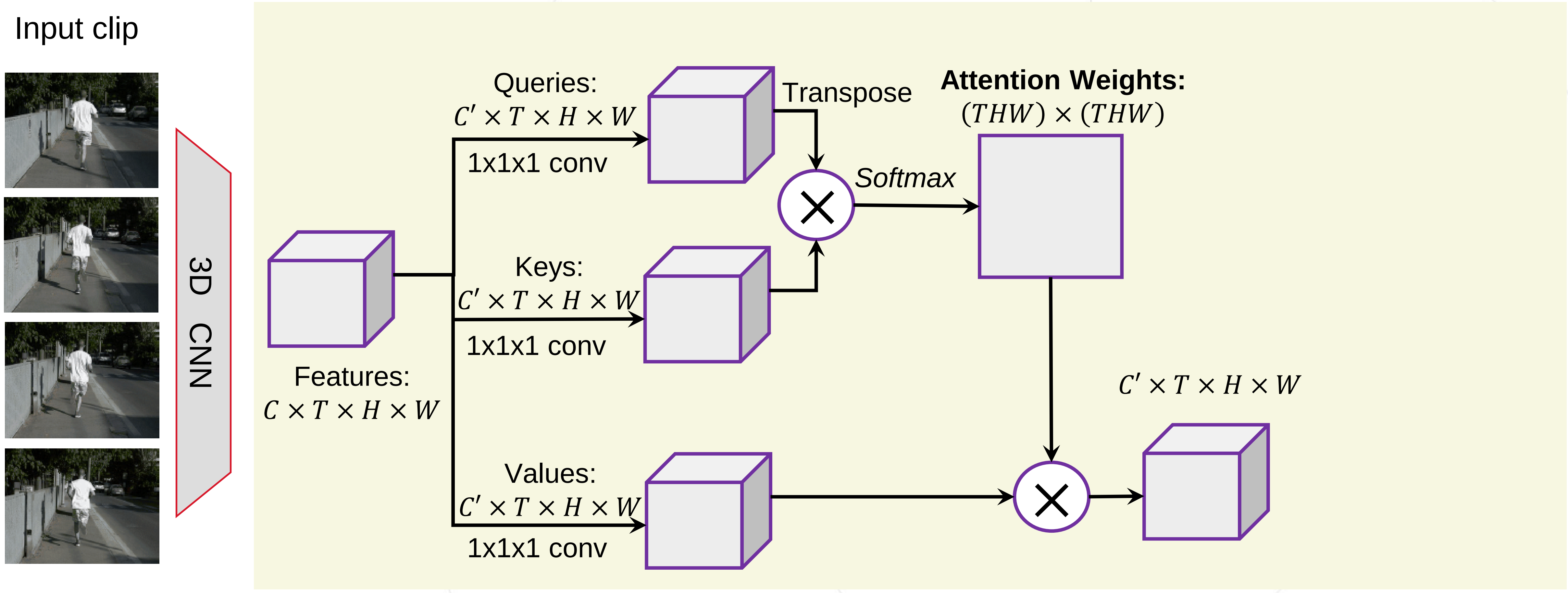
Video classification: Spatio-Temporal Self-Attention (Nonlocal Block)



Nonlocal Block

Based on Slides of Justin Johnson

Video classification: Spatio-Temporal Self-Attention (Nonlocal Block)

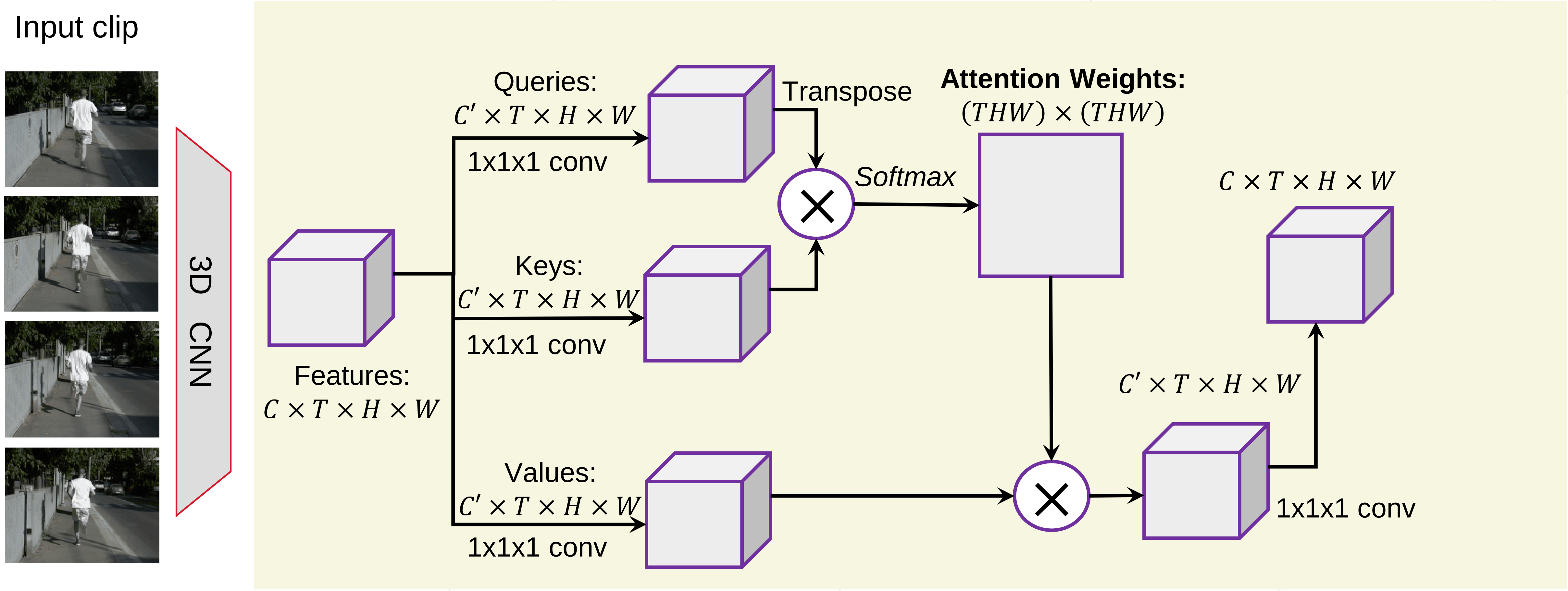


Nonlocal Block

Based on Slides of Justin Johnson

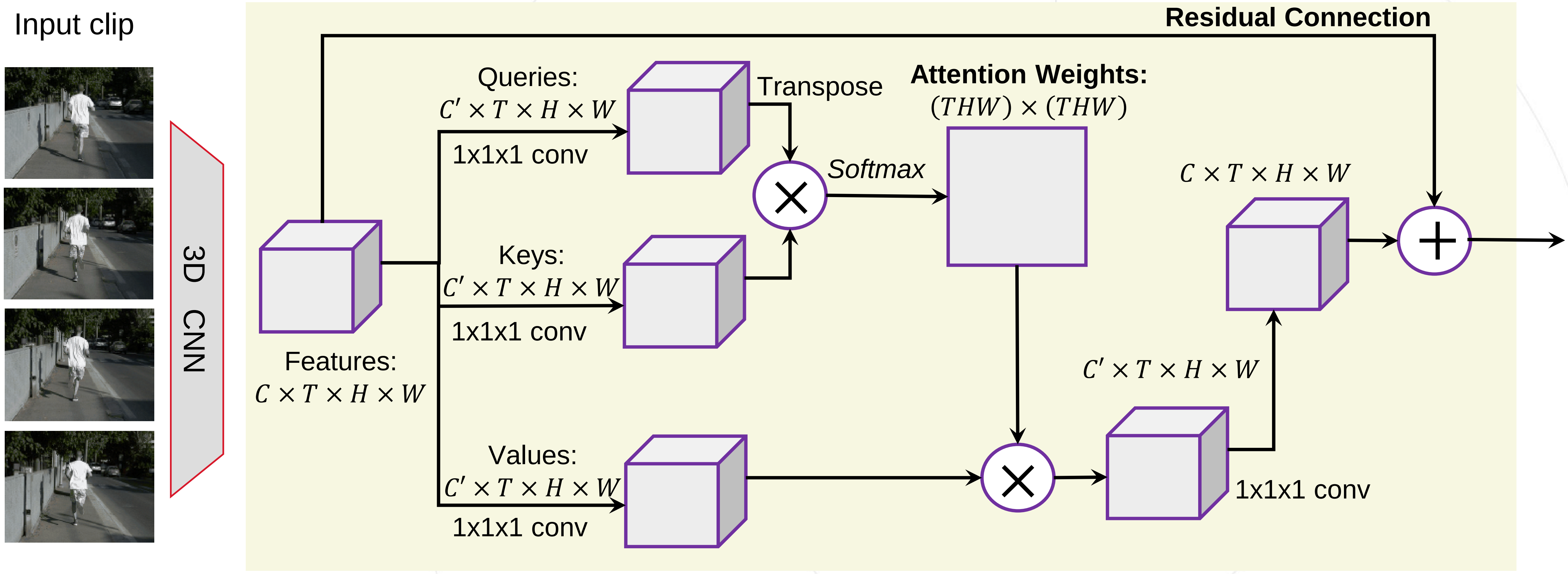
Video classification: Spatio-Temporal Self-Attention (Nonlocal Block)

Input clip



Nonlocal Block

Video classification: Spatio-Temporal Self-Attention (Nonlocal Block)



Nonlocal Block

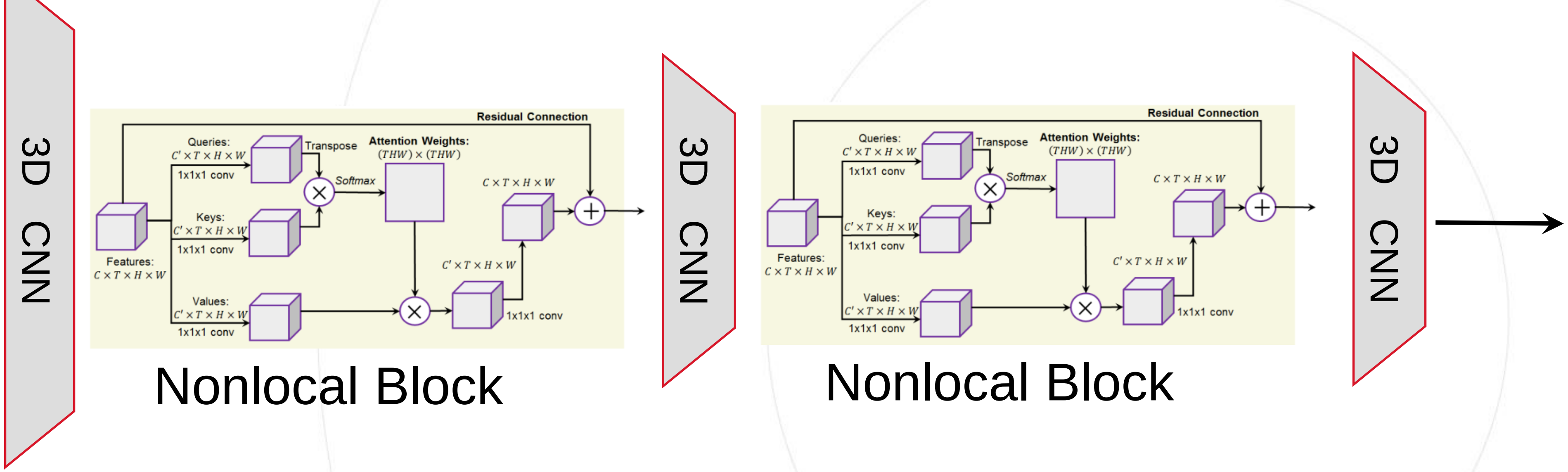
Based on Slides of Justin Johnson

Tsinghua University
sensetime

Video classification: Spatio-Temporal Self-Attention (Nonlocal Block)

Input clip

We can add nonlocal blocks into existing 3D CNN structures.
But what is the best 3D CNN architecture?



Based on Slides of Justin Johnson

Spatio-Temporal Self-Attention (Nonlocal Block)

model	backbone	modality	top-1 val	top-5 val	top-1 test	top-5 test	avg test†
I3D in [7]	Inception	RGB	72.1	90.3	71.1	89.3	80.2
2-Stream I3D in [7]	Inception	RGB + flow	75.7	92.0	74.2	91.3	82.8
RGB baseline in [3]	Inception-ResNet-v2	RGB	73.0	90.9	-	-	-
3-stream late fusion [3]	Inception-ResNet-v2	RGB + flow + audio	74.9	91.6	-	-	-
3-stream LSTM [3]	Inception-ResNet-v2	RGB + flow + audio	77.1	93.2	-	-	-
3-stream SATT [3]	Inception-ResNet-v2	RGB + flow + audio	77.7	93.2	-	-	-
NL I3D [ours]	ResNet-50	RGB	76.5	92.6	-	-	-
	ResNet-101	RGB	**77.7**	**93.3**	-	-	**83.8**

Table 3. Comparisons with state-of-the-art results in **Kinetics**, reported on the val and test sets. We include the Kinetics 2017 competition winner's results [3], but their best results exploited audio signals (marked in gray) so were not vision-only solutions. †: "avg" is the average of top-1 and top-5 accuracy; individual top-1 or top-5 numbers are not available from the test server at the time of submitting this manuscript.

商汤 sensetime

清華大學 Tsinghua University

Outline

Summary

Video classification / action recognition

- Single-frame CNN
- Late fusion
- Early fusion
- 3D CNN
- CNN + RNN
- Two-stream networks
- Self attention