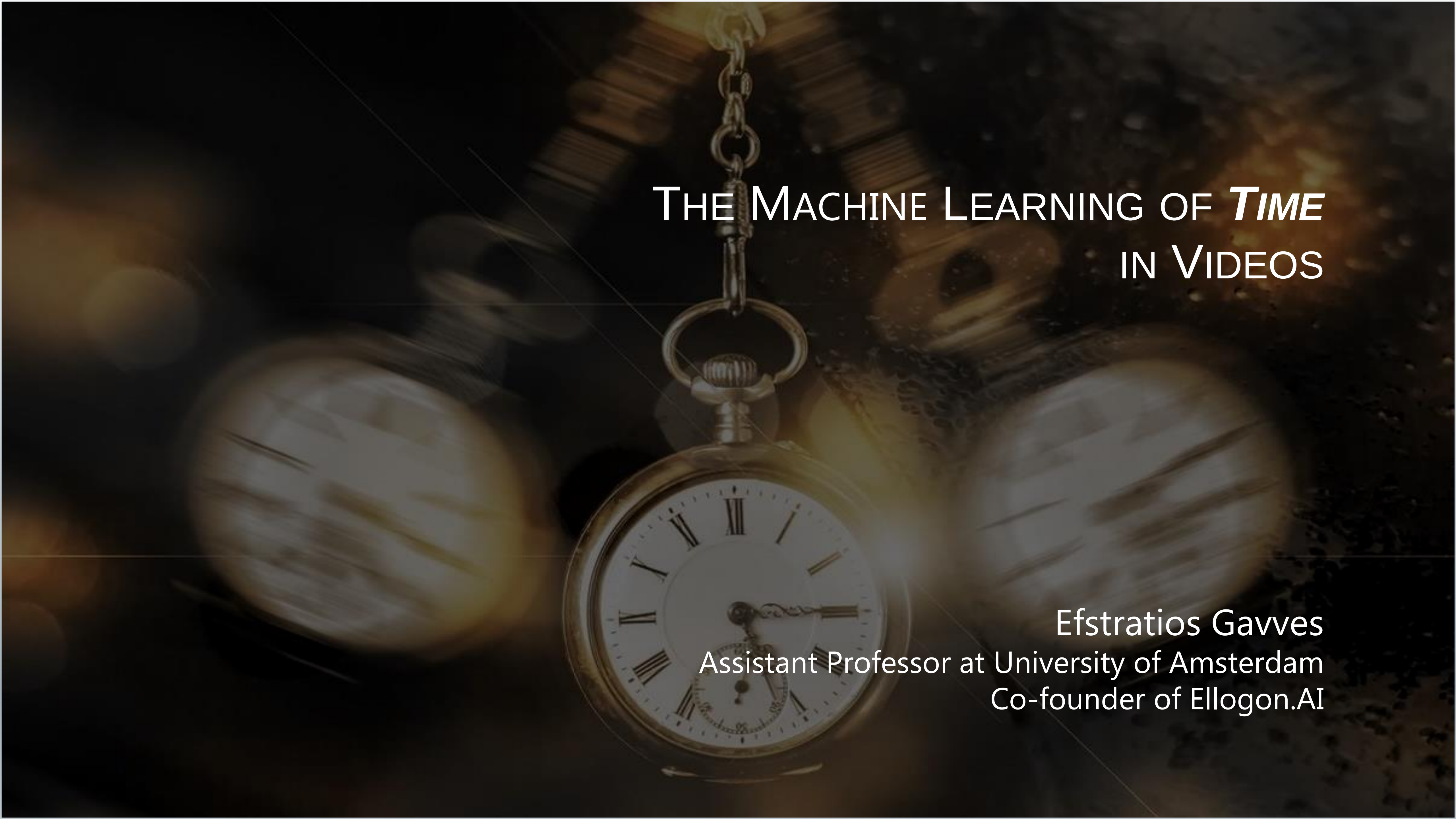


The Machine Learning of ***Time***
in Videos

Efstratios Gavves

Assistant Professor at University of Amsterdam

Co-founder of Ellogon.AI

WHO AM I?

✉ egavves@uva.nl

@egavves

Efstratios Gavves

- Assistant Professor at the University of Amsterdam
  - Scientific Manager at the QUVA Lab
  - QUVA Lab is a joint Academic-Industry Lab between UVA and Qualcomm
  - Teaching Deep Learning (Slides, code available at uvadlc.github.io)
- Co-founder of Ellogon.AI
  - Machine Learning for Clinical Trials and Pharmaceutical Design
  - Partnering up with the Dutch National Cancer Institute against oncology
    - One of the biggest research centers worldwide with huge data
  - If interest, please come find me

UNIVERSITY OF AMSTERDAM

QUVA Deep Vision Lab

ELLOGON.AI

Video Modelling Today: Short

- Spatiotemporal Encoders: convolve up to a few dozen frames

- Action Classification: process up to few seconds

- Efficient Video Models: don’t really exist

- Self-supervised Learning: predicting immediate spatio-temporal context

Video Modelling Tomorrow: Long

- Spatiotemporal Encoders: thousands of frames
- Sequence Learning of Complex Actions: dozens of minutes or hours long
- Efficient Video Models: scaling up cannot be done without contemplating efficiency
- Self-supervised Learning: from spatio-temporal context to temporal properties

Video Temporal Modelling of tomorrow about encoding transitions over long term and dynamics …
… instead of encoding short spatio-temporal (static) patterns

Video Dynamics Learning

- When it comes to long or streaming videos the important questions are:

Is there a difference between a video sequence and other types of sequences?
What are the meaningful dynamics of the video content and how to capture them?
How to encode the meaningful dynamics in a “non-catastrophic forgetting” manner?
How to encode multiple temporal complexities of dynamics?

Can we design video specialized models and architectures for dynamics?
Not models that extend our favorite 2D convnet

VIDEOLSTM

- VideoLSTM convolves, attends and flows for action recognition, CVIU 2018
  - Code: https://github.com/zhenyangli/VideoLSTM

Zhenyang Li

Efstratios Gavves

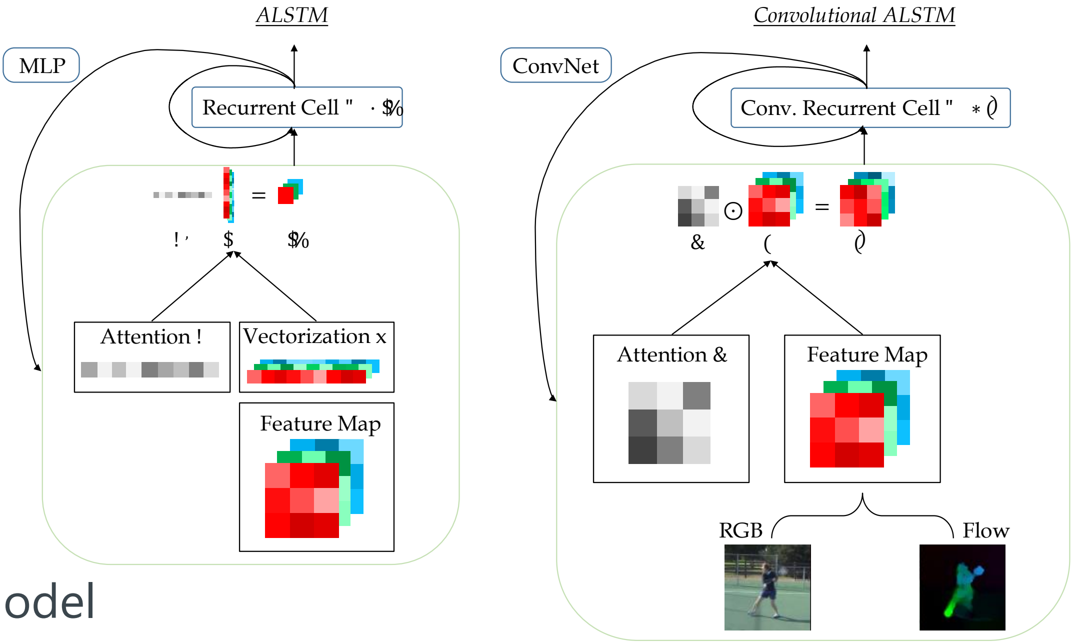
Mihir Jain

Cees Snoek

VideoLSTM: TL;DR

- LSTM relies on inner products
  - Equivalent to translation-variant fully Connected MLPs
  - Why not replace all operations with convolutions?

- Attention in LSTMs typically on RGB inputs
  - What moves is what acts
  - Why not use motion just for the attention?



- VideoLSTM proposes a Convolutional A(ttention) LSTM model
  - The video encoding using RGB channels
  - The attention encoding using motion channels

Convolutional (A) LSTM

- Replace the fully connected multiplicative operations in an LSTM unit with convolutional operations

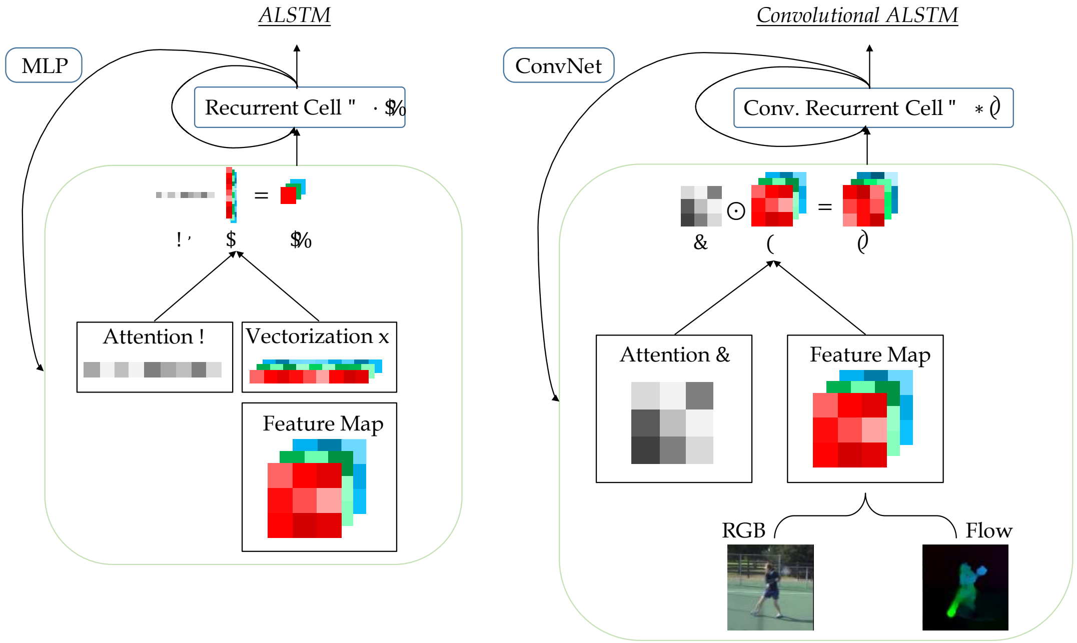
$$
\begin{aligned}
I_t &= \sigma(W_{xi} * \widetilde{X}_t + W_{hi} * H_{t-1} + b_i)\\
F_t &= \sigma(W_{xf} * \widetilde{X}_t + W_{hf} * H_{t-1} + b_f)\\
O_t &= \sigma(W_{xo} * \widetilde{X}_t + W_{ho} * H_{t-1} + b_o)\\
G_t &= \tanh(W_{xc} * \widetilde{X}_t + W_{hc} * H_{t-1} + b_c)\\
C_t &= F_t \odot C_{t-1} + I_t \odot G_t\\
H_t &= O_t \odot \tanh(C_t),
\end{aligned}
$$

- Generate attention by shallow ConvNet instead of MLP

$$Z_t = W_z * \tanh(W_{xa} * X_t + W_{ha} * H_{t-1} + b_a)$$

$$A_t^{ij} = p(att_{ij}|X_t, H_{t-1}) = \frac{\exp(Z_t^{ij})}{\sum_i \sum_j \exp(Z_t^{ij})}$$

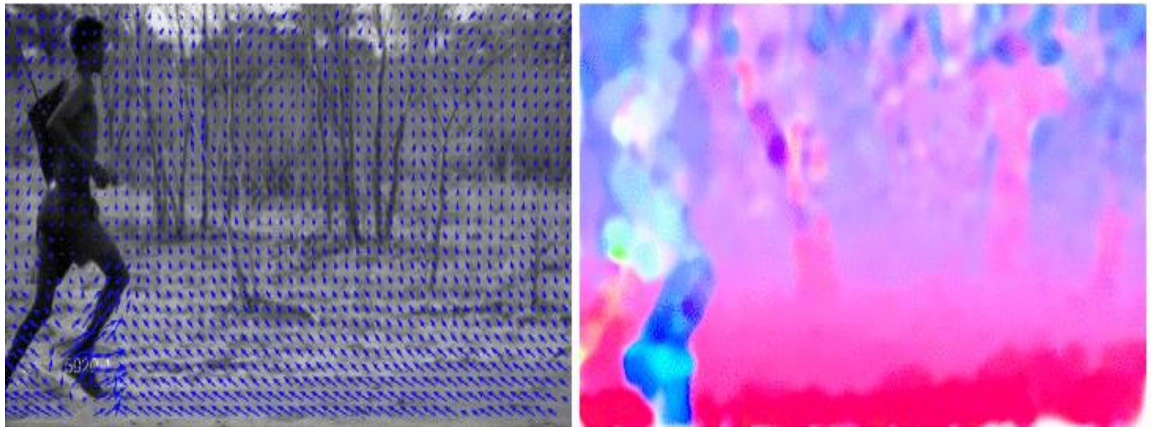
$$\widetilde{X}_t = A_t \odot X_t$$

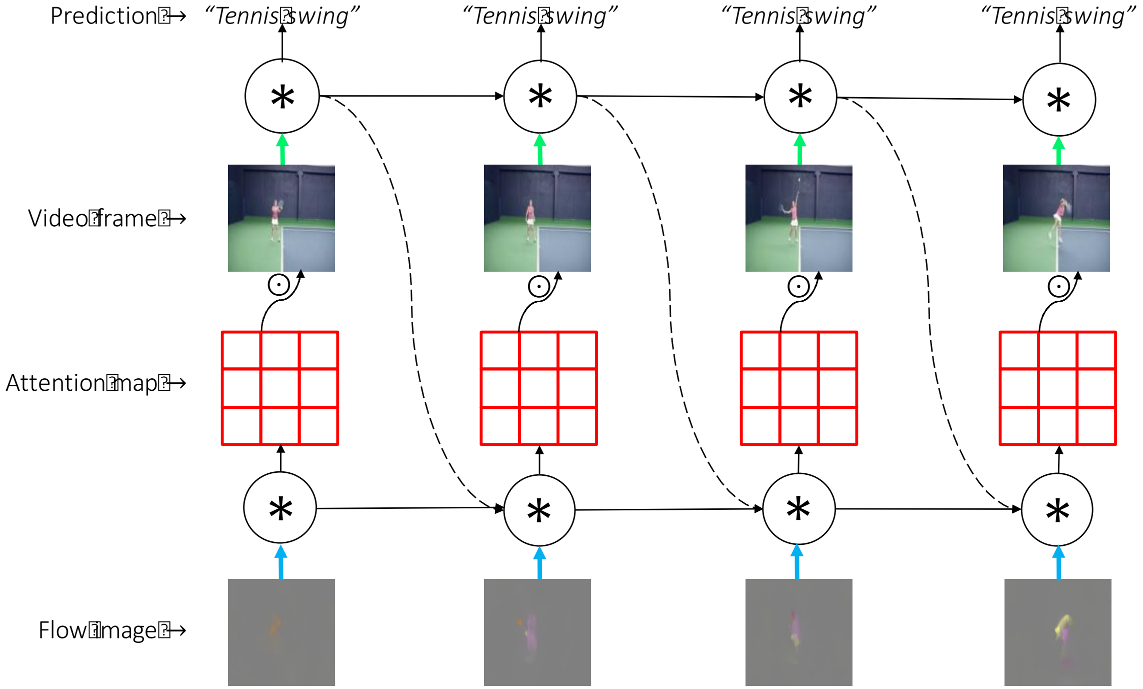


Convolutional ALSTM preserves spatial dimensions over time

Motion-Based Attention

- Motion offers crucial clue where to attend in video

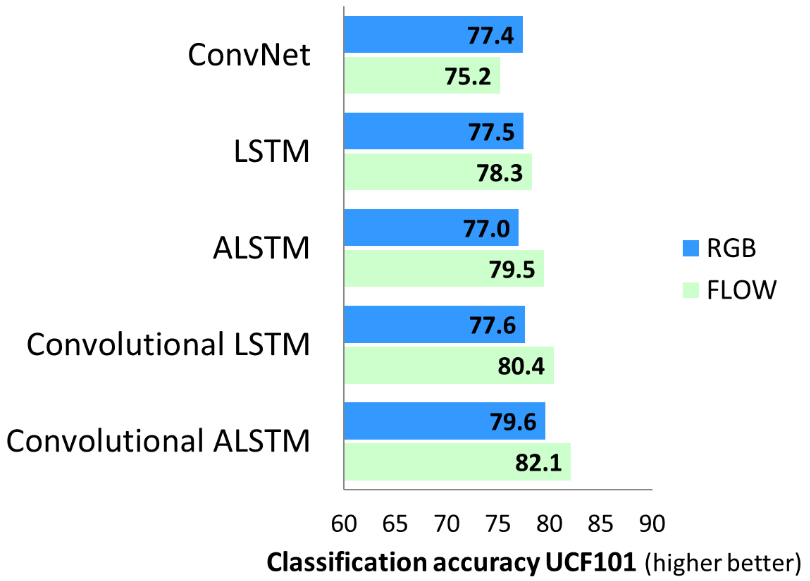




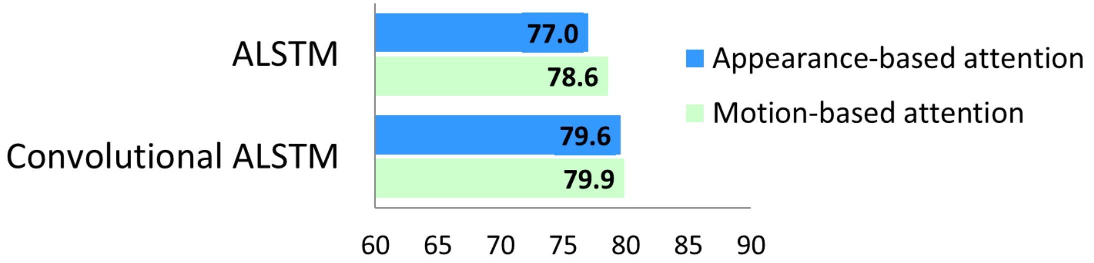
Motion information to infer the attention in each frame

EXPERIMENTS

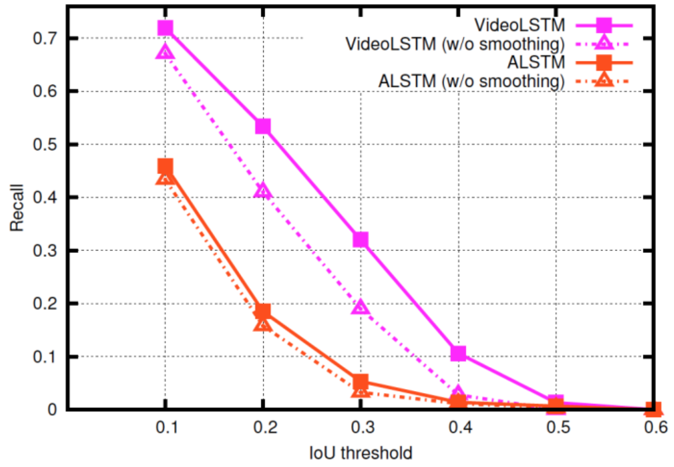
Convolutions + Attention makes sense!



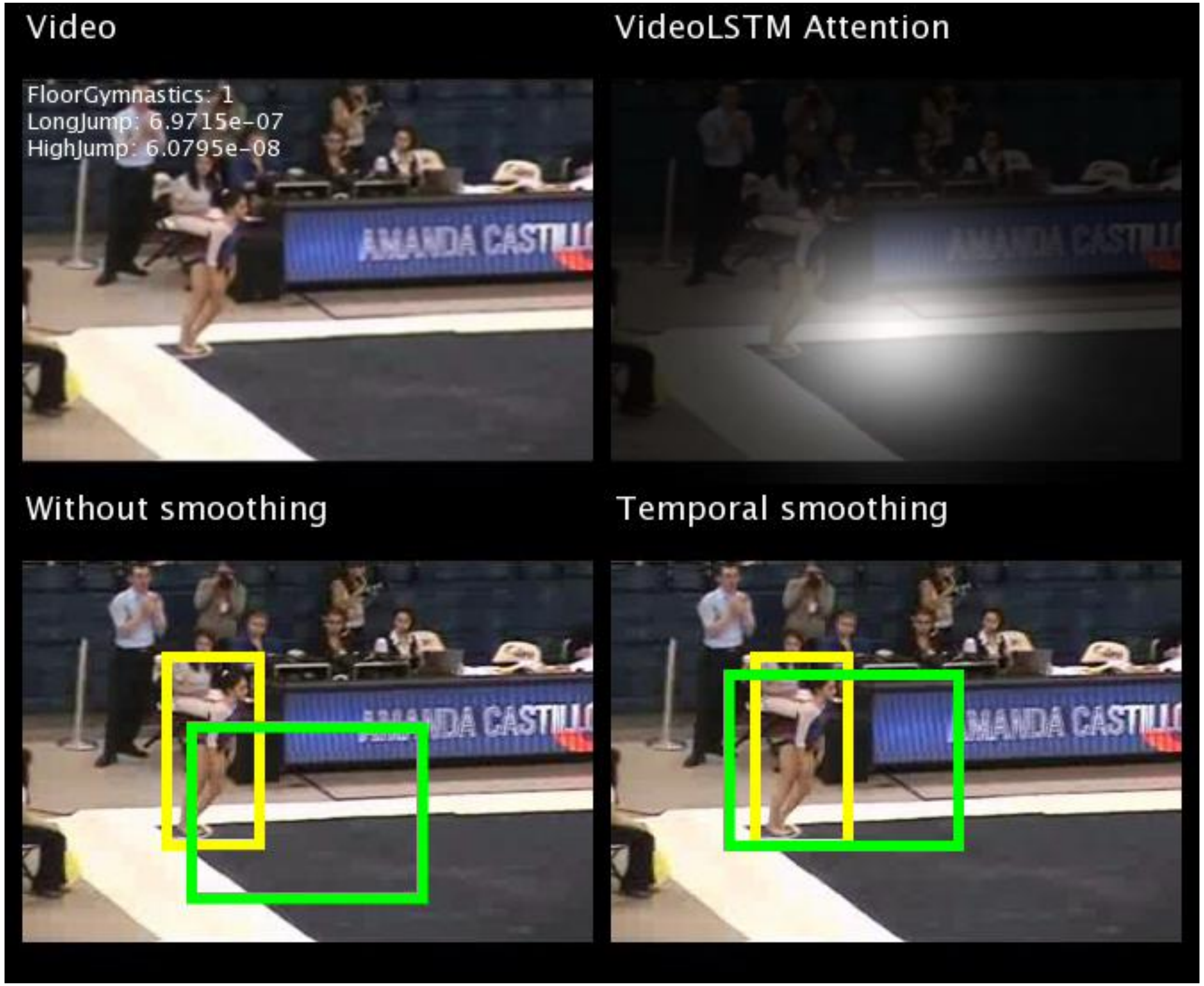
Motion for Attention makes sense!



Localization for free



Qualitative Results



VideoLSTM: What Have We Learned?

- Hardwiring convolutions in attention LSTM

- Derives attention from what moves in video

- Leads to a promising and well performing video-unique deep architecture

- Localization from a video-level action class label only

VIDEOLSTM: OPEN QUESTION

Does LSTM really encode sequential dynamics?
Or does it simply perform some sort of pooling?

VIDEOTIME

- Video Time: Properties, Encoders and Evaluation, BMVC 2018
  - Code: https://github.com/QUVA-Lab/

Amir Ghodrati

Efstratios Gavves

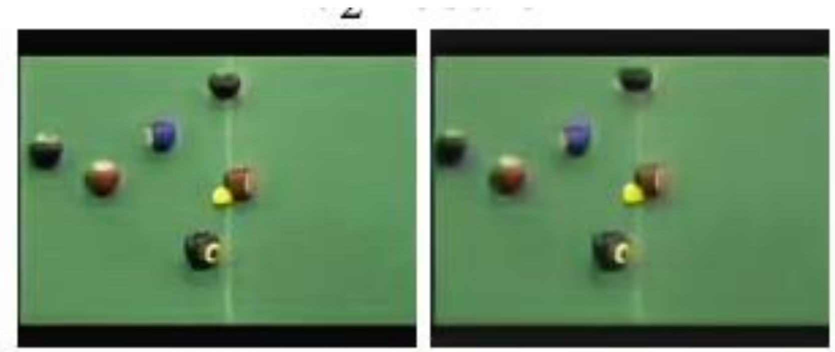
Cees Snoek

VIDEOTIME: TL;DR

- What is the contribution of modeling time in video tasks?
  - Considering video as a sequence, do sequence models like LSTMs really encode temporal dynamics?

- What does it even mean “Encode Temporal Dynamics”?
  - Investigate properties of times in videos for which time is the modifier

- VideoTime proposes Time-Aligned DenseNets
  - Much better temporal encoders!!



PLAYING WITH TIME

A or B?

All of Them are In Reverse

A or B?

(Some) Properties of Time in Videos

- There is a clear distinction between the forward and the backward arrow of time

Temporal Asymmetry

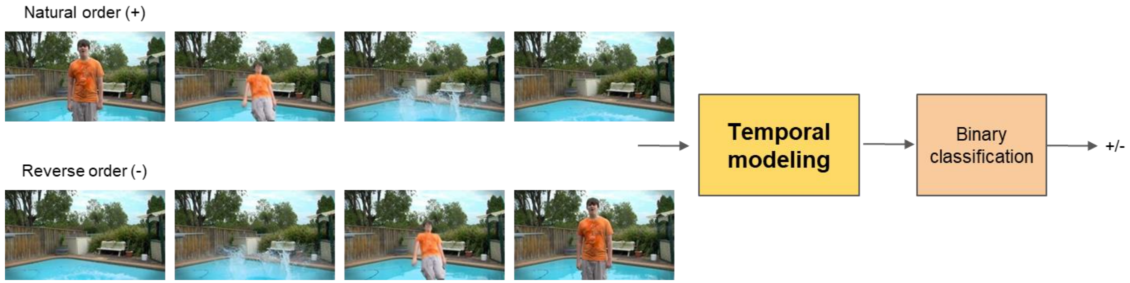
Temporal Causality

Temporal Continuity

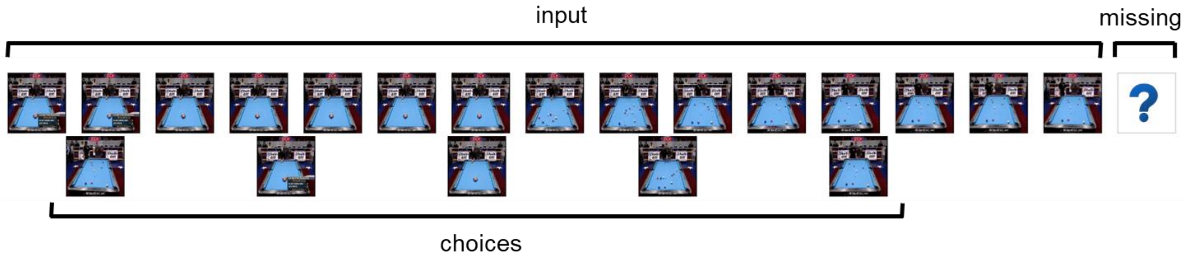
Temporal Redundancy

How to Quantify These Properties?

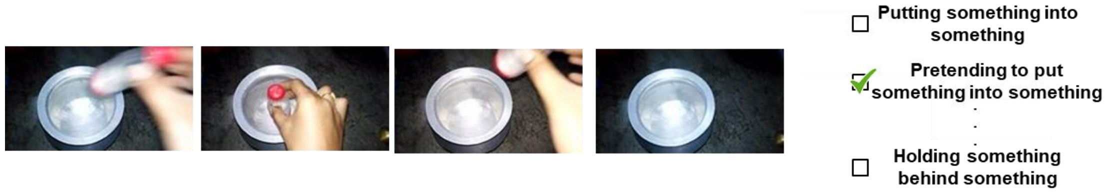
- Temporal asymmetry → Arrow of time prediction



- Temporal continuity → Future Frame Selection

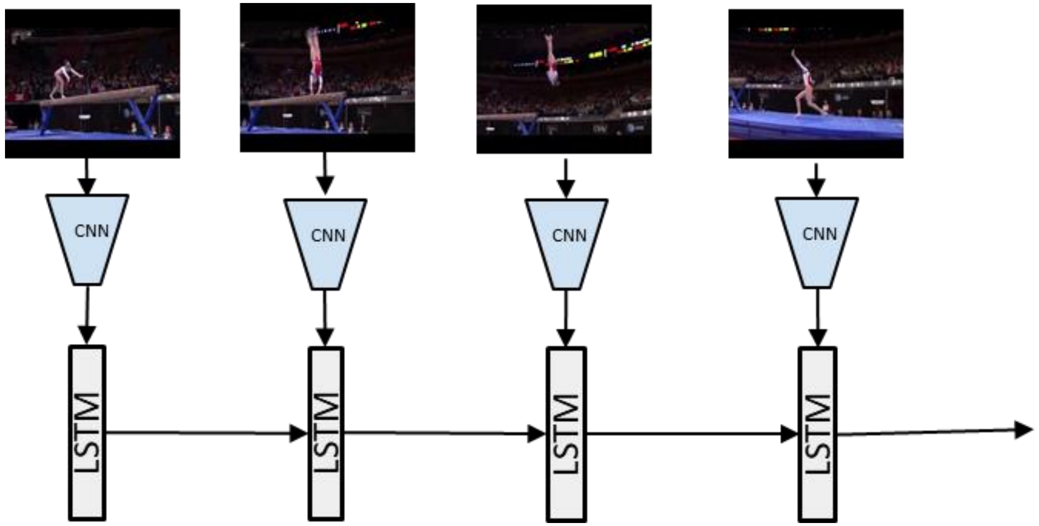


- Temporal causality → Action Template Classification


□ Putting something into something
✓ Pretending to put something into something
□ Holding something behind something

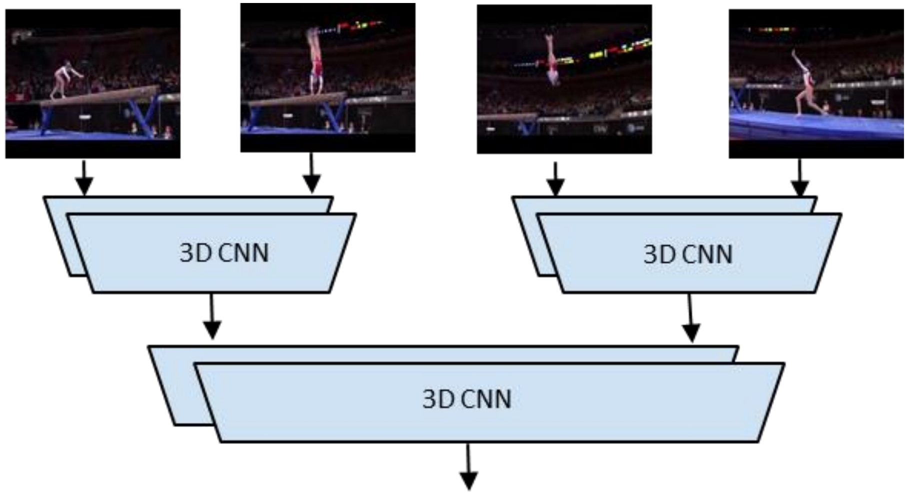
TWO DOMINANT APPROACHES

LSTMs learn transitions between subsequent states



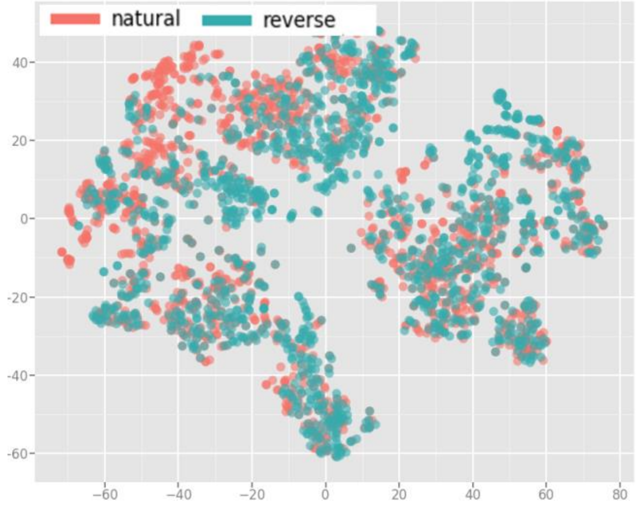
Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. Neural computation, 1997

3D convolutions learn spatiotemporal correlations

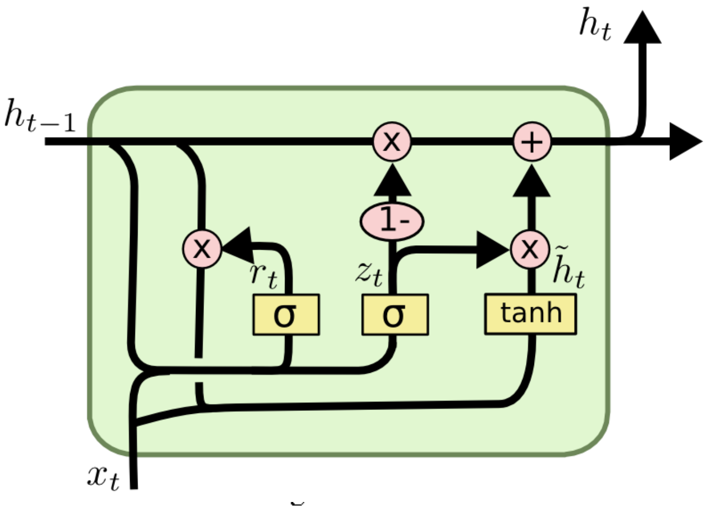


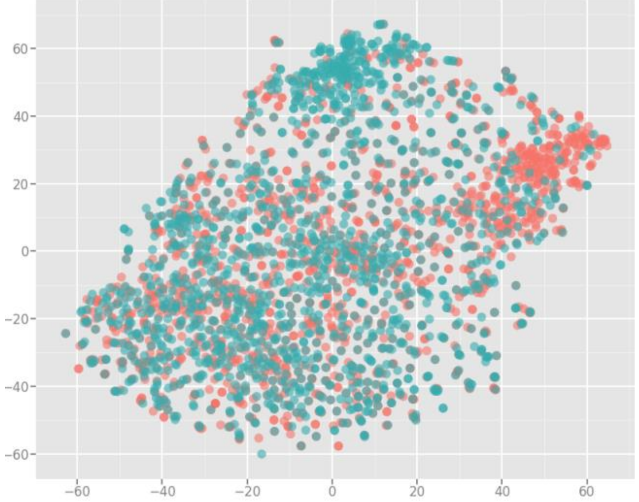
Ji et al. 3d convolutional neural networks for human action recognition. PAMI, 2013
Tran et al., Learning Spatiotemporal Features with 3D Convolutional Networks, ICCV 2015

LSTM and C3D: Arrow of Time?

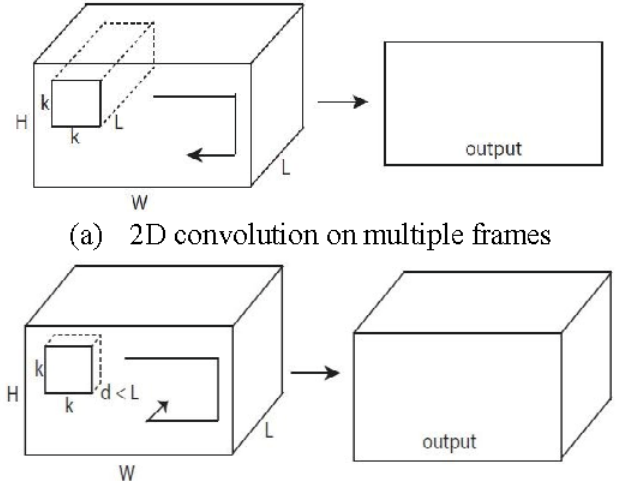


LSTM





C3D



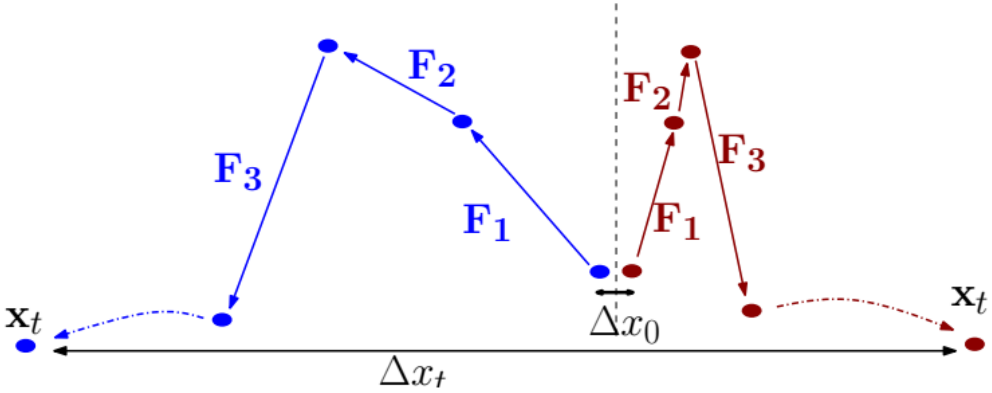
(a) 2D convolution on multiple frames

(b) 3D convolution on multiple frames

Revisiting Recurrent Neural Networks

- Recurrent Nets are highly sensitive dynamical systems (Pascanu, 2013)
  - Even considering highly discriminative one-hot vector inputs
  - Gradients very sensitive to initialization → Poor learning! → No generalization

- Visual features over time -even the best ones- are:
  - <u>much noisier</u>
  - <u>much less discriminative</u>
  - <u>much more redundant</u>



- Learning LSTM on videos is orders of magnitude harder
  - Chaotic regime → no useful gradients → absolutely no useful learning
  - Forward and Backward LSTM score the same accuracy on arrow of time

Basically, with high-dim noisy inputs LSTMs do not do sequence modelling but some weird entangled pooling

PROPOSAL: TIME-ALIGNED DENSENET

- ConvNets are much better with vanishing and exploding gradients, noisy and redundant inputs

Hypothesis

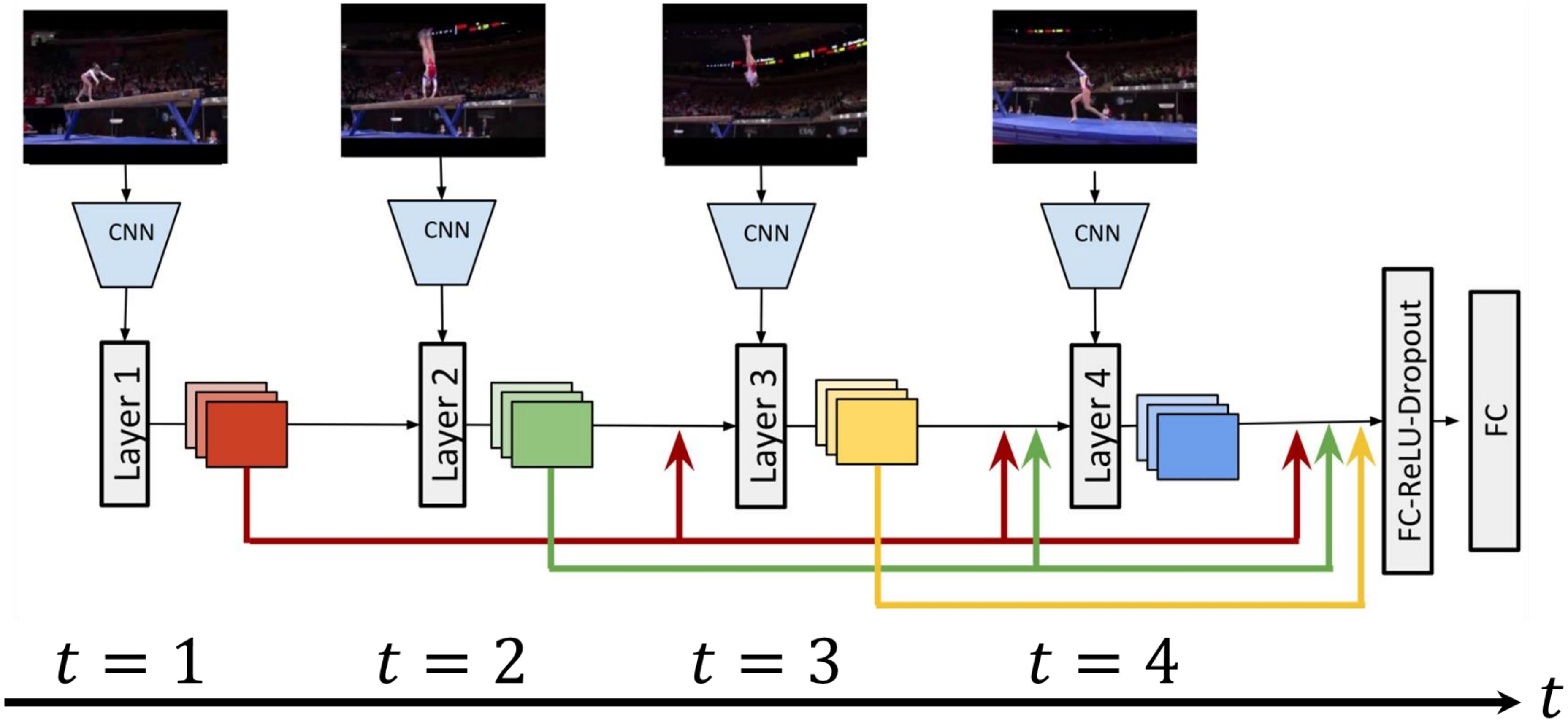
ConvNets can handle vanishing/exploding/noisy/redundant because they do not share parameters.

- No parameter sharing → no chaotic regime
- Moreover, the premise of LSTM parameter sharing is infinite Markov chains
- In practice, however, we chop it off at T steps → like a ConvNet with T layers

- Idea: Why not flip the ConvNet to align the layers with time steps?

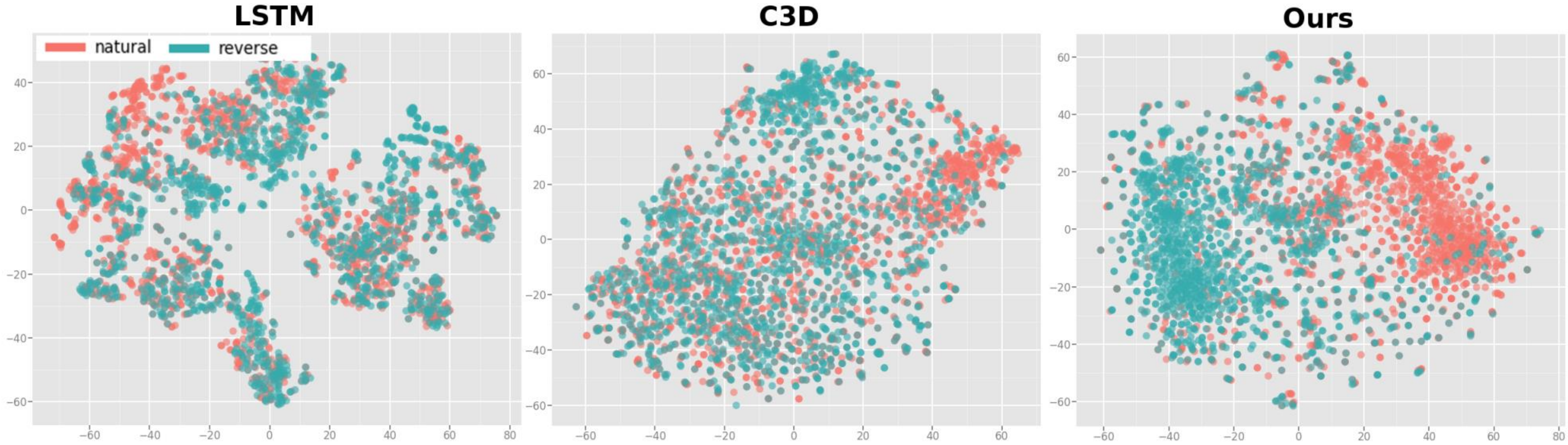
Proposal: Time-aligned DenseNet

- Idea: Why not flip the ConvNet to align the layers with time steps?
- No vanishing/exploding gradients, no problems with noisy and redundant inputs



Rechecking Arrow of Time

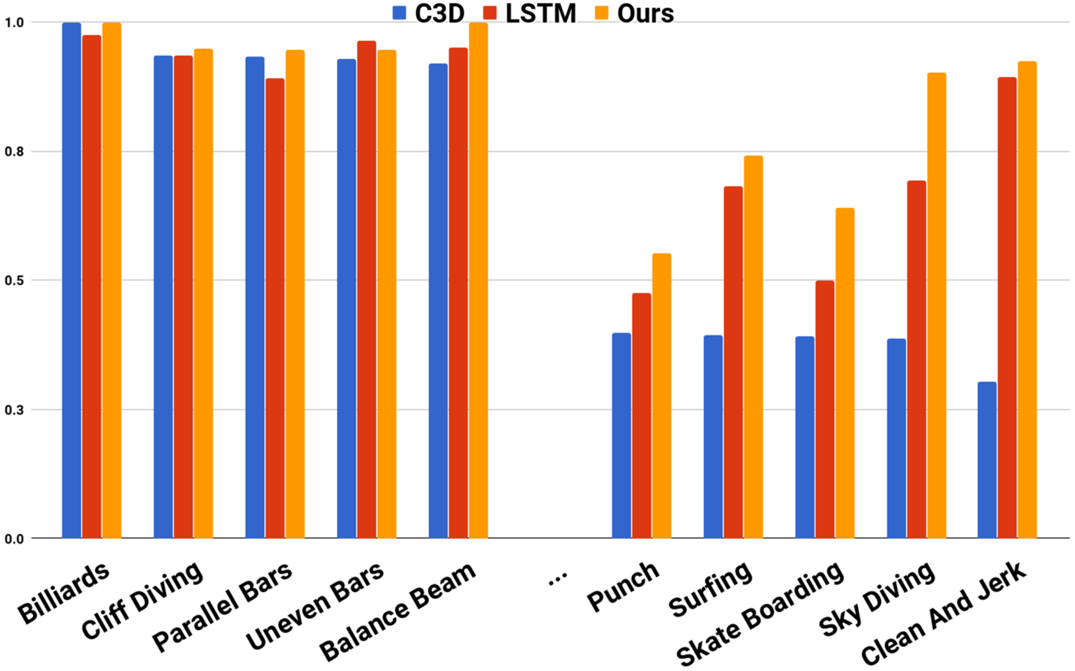
- Time-Aligned DenseNet gives much cleaner temporal clusters



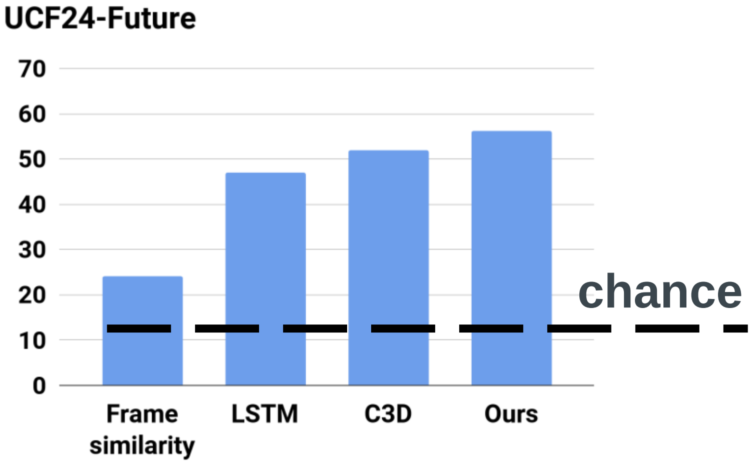
Conclusion: Poor temporal modelling is likely due to hard –and thus unsuccessful- optimization

EXPERIMENTS

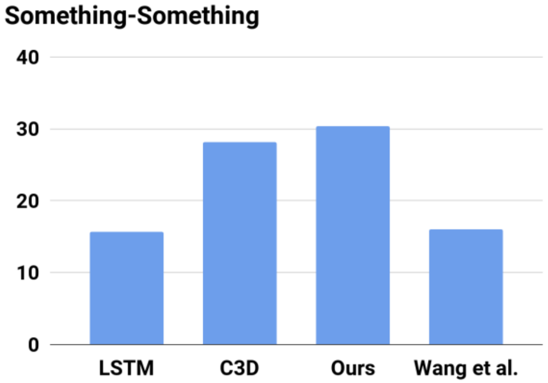
Arrow of time: improved temporal asymmetry
Especially for temporally causal classes
LSTM better than C3D



Future frame: improved temporal continuity
Especially for temporally causal classes
C3D better than LSTM



Action Templates: improved temporal causality
C3D better than LSTM
Sometimes, correlation implies causation :P



VideoTIME: What Have We Learned?

- Poor temporal modelling is likely due to hard –and thus unsuccessful- optimization

- As the complexity of a task increases, spatiotemporal correlation learning methods like C3D performs better than transition-based learning methods like LSTM

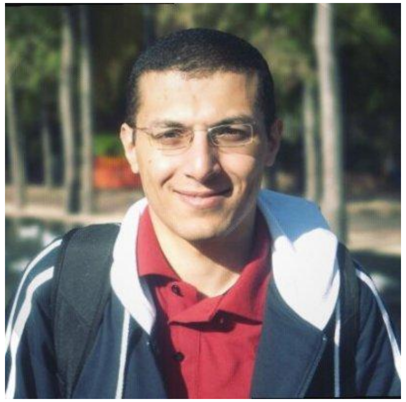
- Time-aligned DenseNet performs better than LSTM mostly due to shared parameterization of LSTMs

VideoTIME: Open Question

Sure, we can model time better. So what?
What about using it for strong self-supervised learning?
Maybe time is more important in modelling & recognizing complex actions?

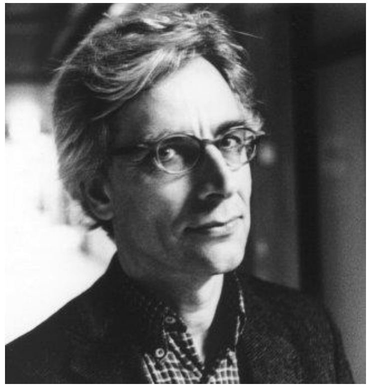
TIMECEPTION

- VideoLSTM convolves, attends and flows for action recognition, CVIU 2019 (Oral on Tuesday)
  - Code: https://github.com/noureldien/timeception



Noureldien Hussein

Efstratios Gavves



Arnold Smeulders

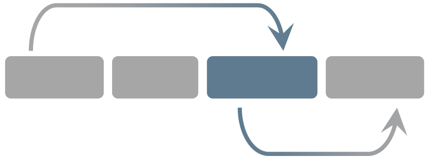
TIMECEPTION: TL;DR

- Most video methods today focus on few second videos
  - Is this realistic? What happens with minutes-long, hours-long or even streaming videos?

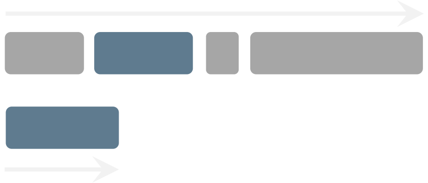
- What does it even mean “Complex action”?
  - Investigate properties of complex actions over long time videos

- Timeception
  - Can scale up to dozens of minutes without a sweat at high accuracies

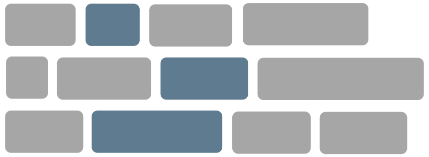
1. Dependency



2. Long-range



3. Temporal Extent



Problem Complex Actions

Problem Complex Actions

Preparing Breakfast

Complex Action

Stirring Food

One-action

Problem Complex Actions

Preparing Breakfast

Complex Action

1. Long-range
2. Temporal Extent
3. Temporal Dependency

Problem 1. Long-range

● get ● cook ● put ● wash

One-action ~2 sec.

Problem 1. Long-range

● get ● cook ● put ● wash

One-action ~2 sec.

Complex Action ~30 sec.

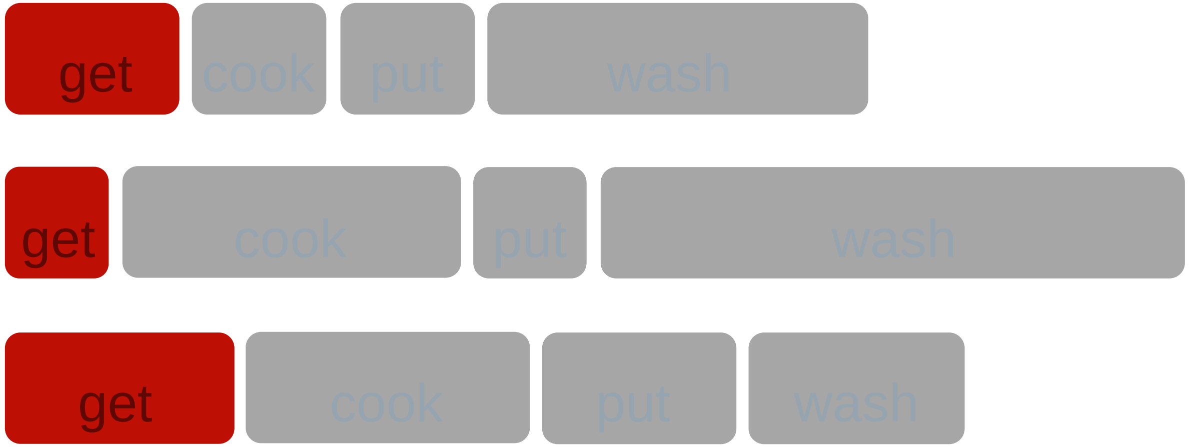
Problem 2. Temporal Extent

● get

● cook

● put

● wash



get cook put wash

get cook put wash

get cook put wash

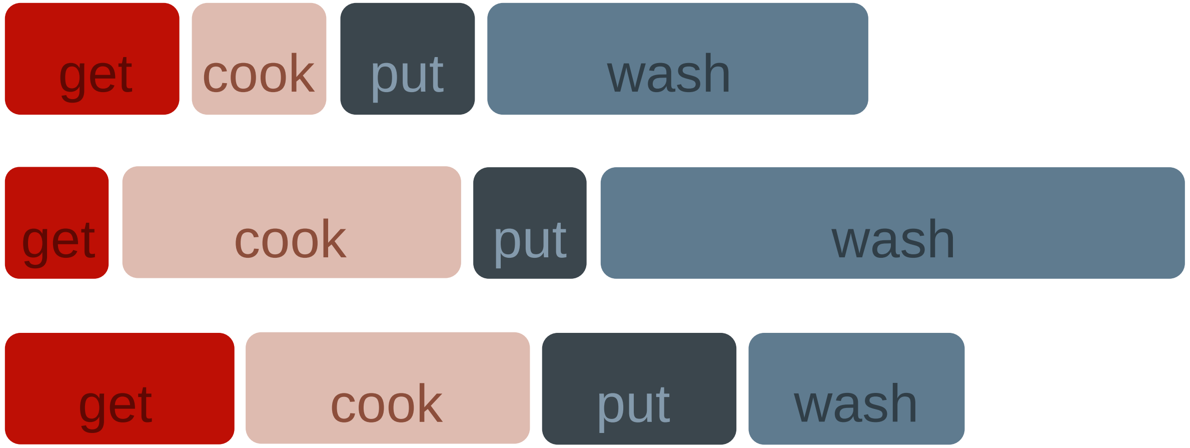
Problem 2. Temporal Extent

● get

● cook

● put

● wash



Problem 3. Temporal Dependency

● get

● cook

● put

● wash

get cook put wash

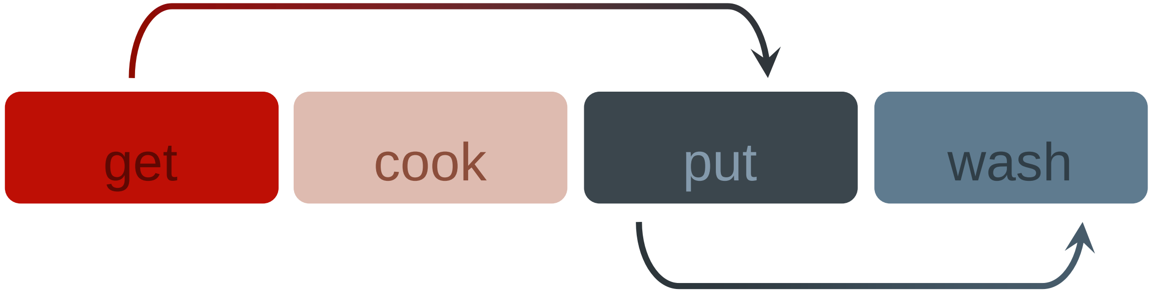
Problem 3. Temporal Dependency

● get

● cook

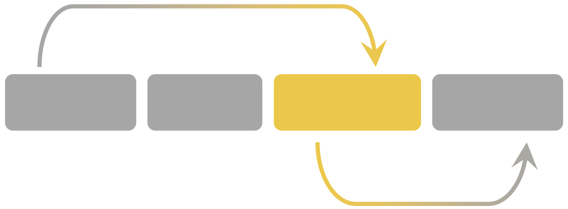
● put

● wash

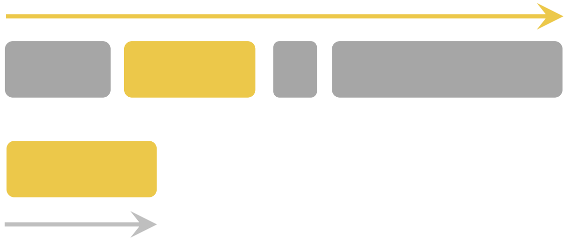


Problem Complex Actions

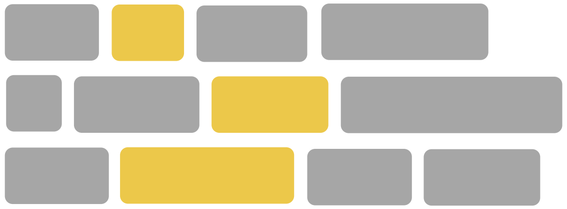
1. Dependency



2. Long-range



3. Temporal Extent



Problem

Design a model addressing all three properties?

Decomposition of convolutional operations the only way forward

But how can we make it permissible for minute long videos?

We note that all convolution decompositions are effectively chain subspace projections

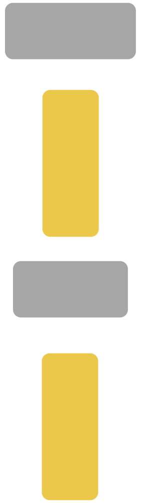
$$\boldsymbol{w} \propto \boldsymbol{w}_{\alpha} * \boldsymbol{w}_{\beta} * \boldsymbol{w}_{\gamma} * \cdots$$

The order in the chain should not be really that important

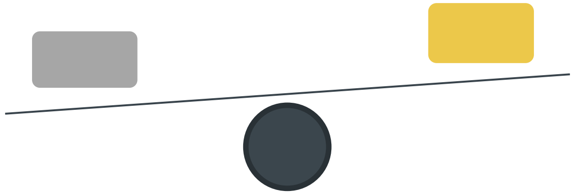
Problem

Subspace projections: Design Principles

1. Subspace modularity



2. Subspace balance



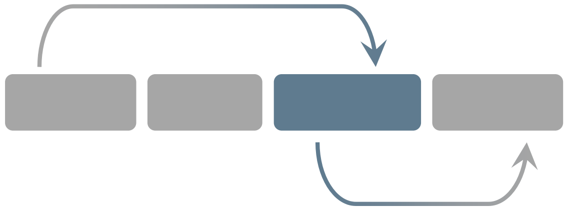
3. Subspace efficiency

METHOD

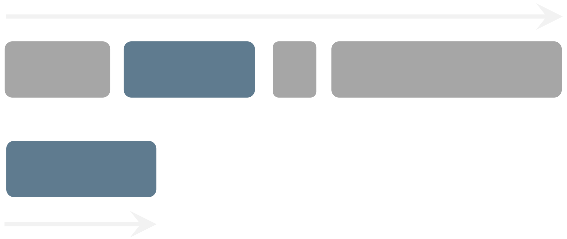
Method Timeception

✓

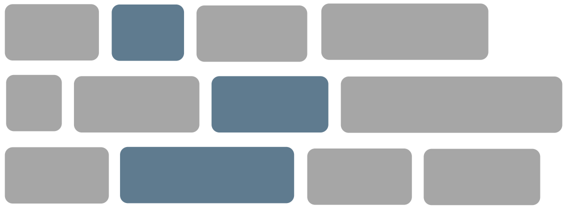
1. Dependency

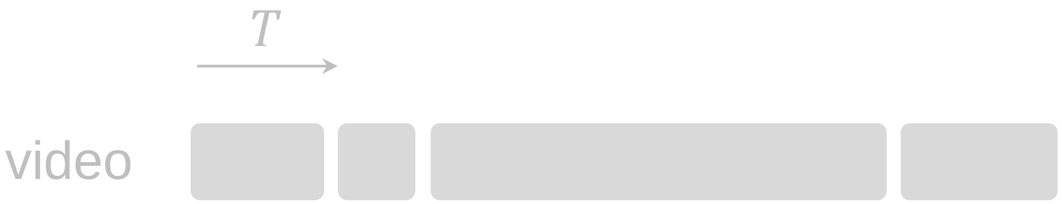


2. Long-range



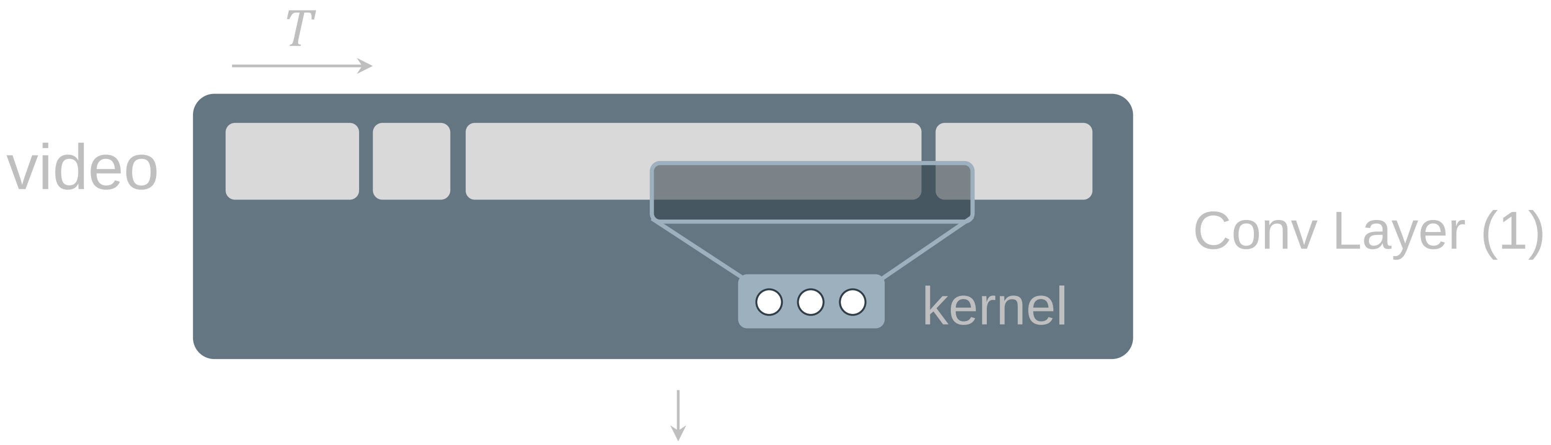
3. Temporal Extent

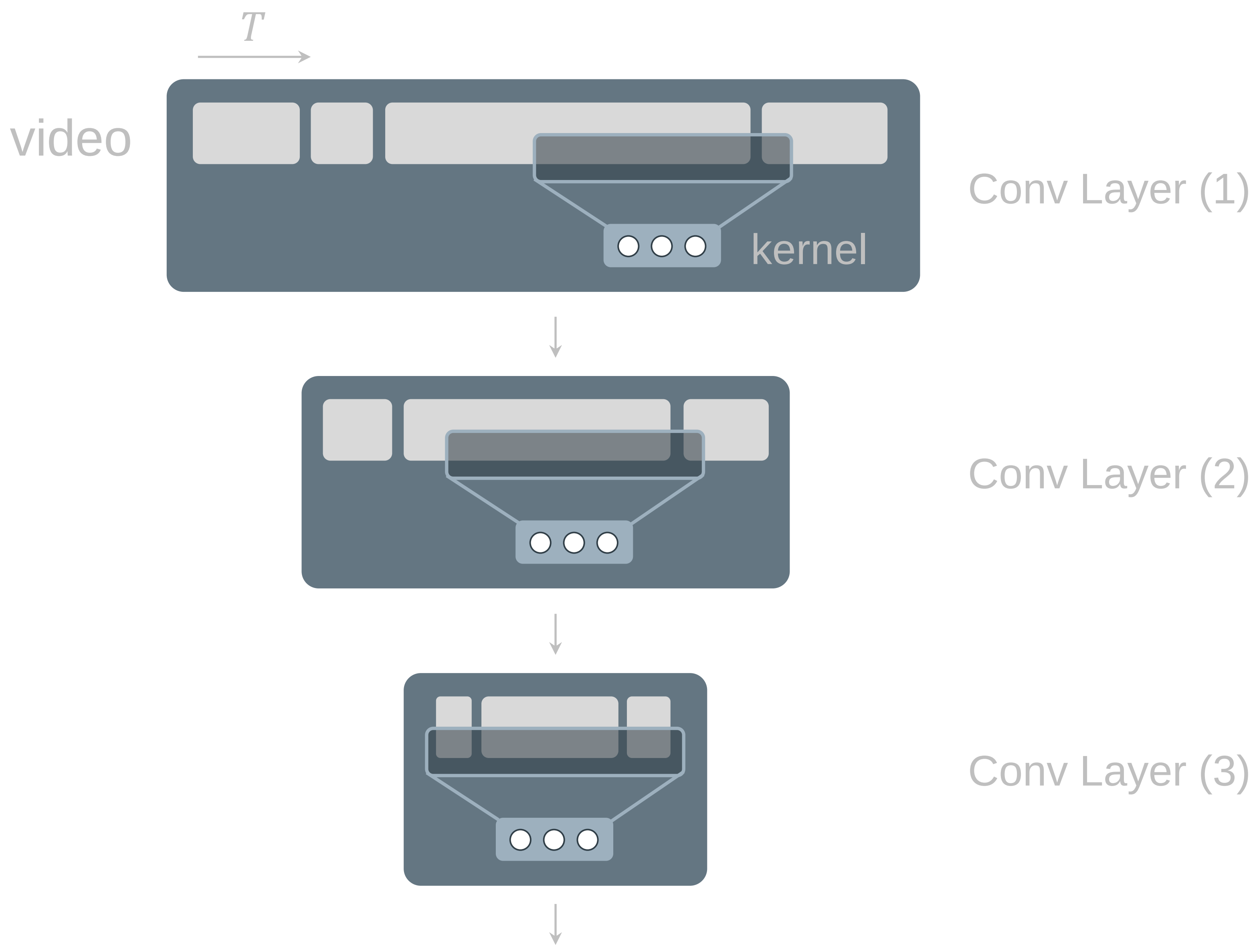


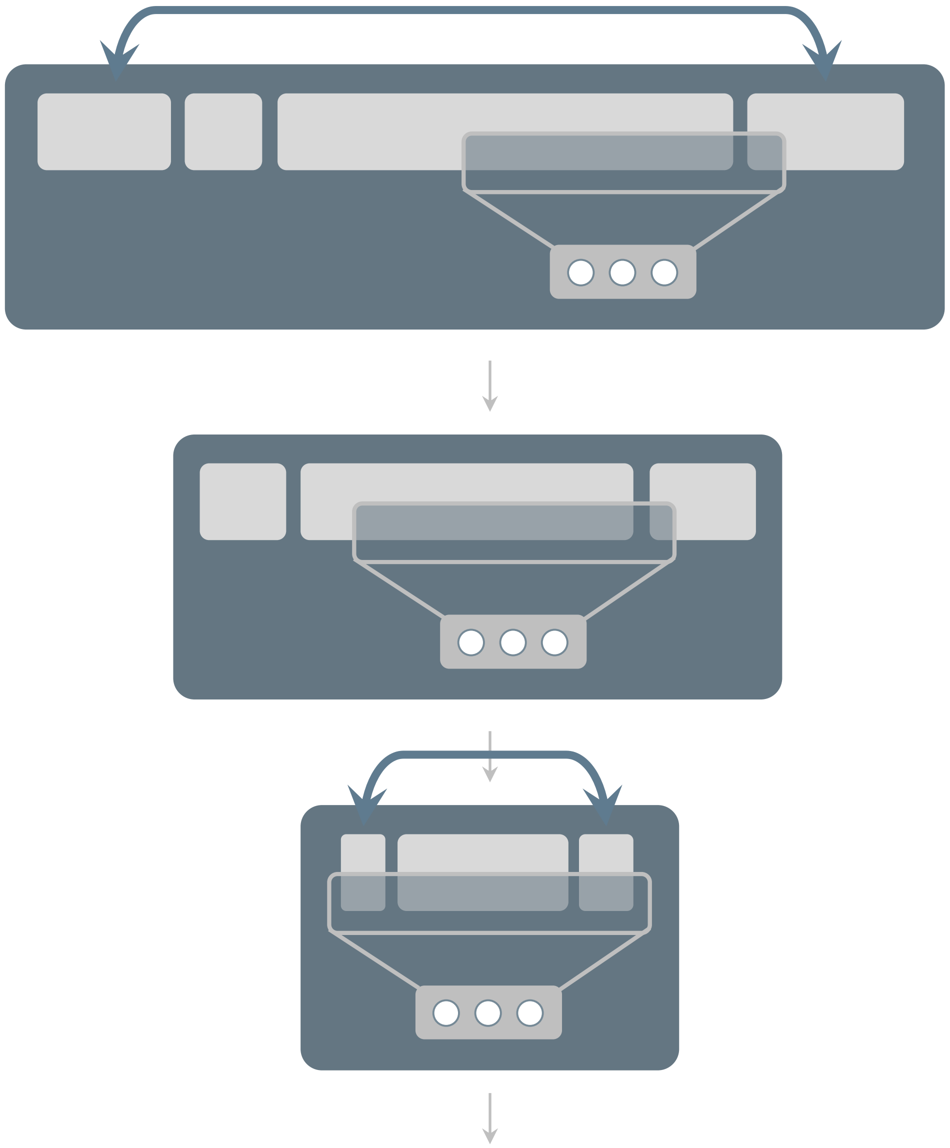


T

video


T
video
kernel
Conv Layer (1)


T
video
kernel
Conv Layer (1)
Conv Layer (2)
Conv Layer (3)

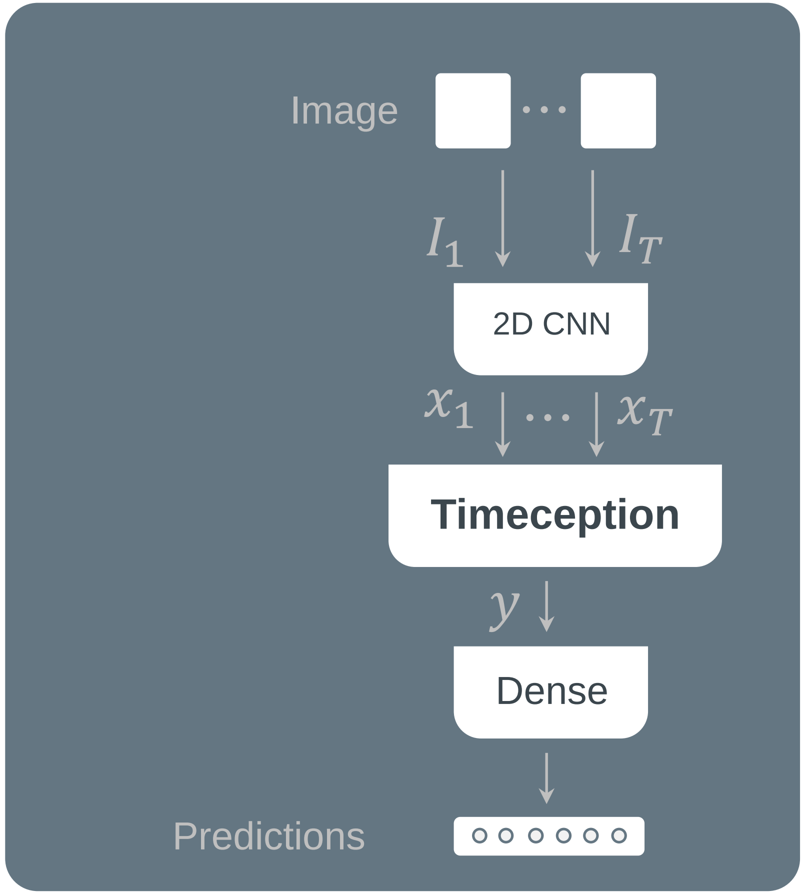


Conv Layer (1)

Conv Layer (2)

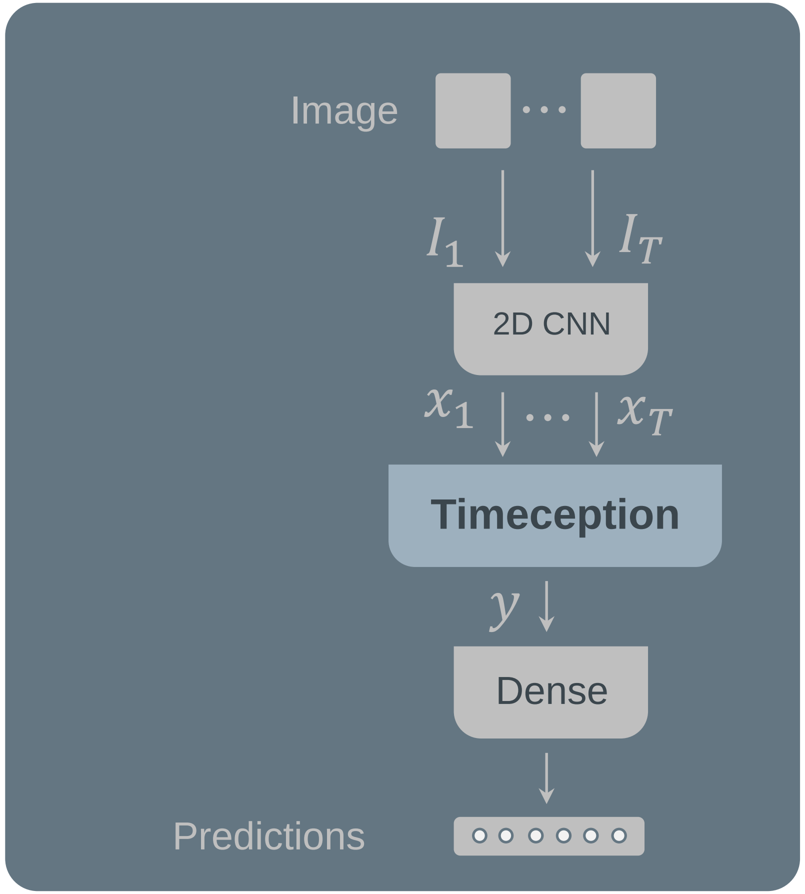
Conv Layer (3)

Method Timeception



Model Overview

Method Timeception

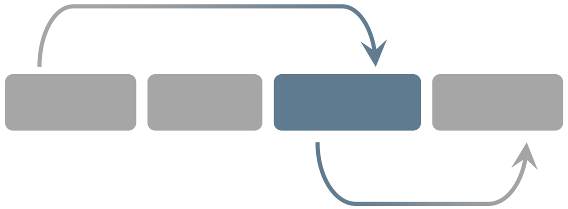


Model Overview

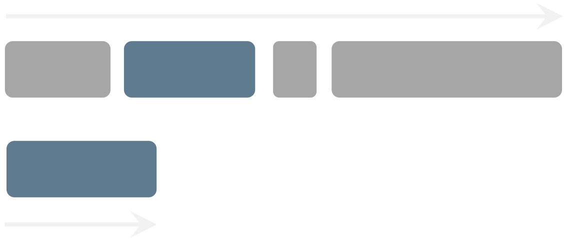
Method Timeception

✓

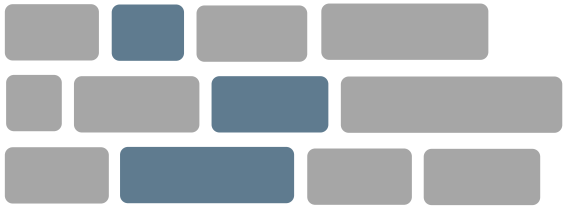
1. Dependency



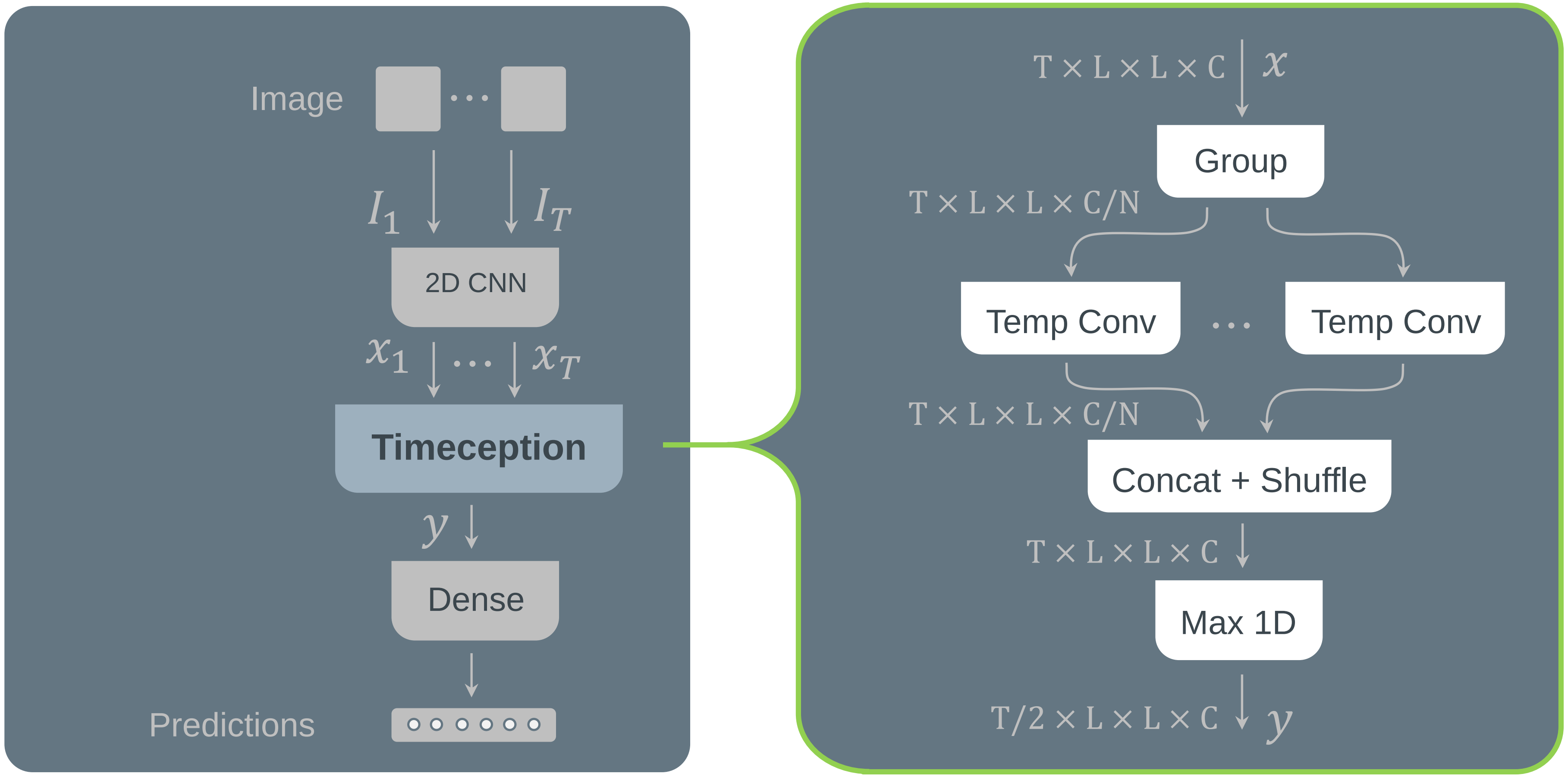
2. Long-range



3. Temporal Extent



Method Efficiency



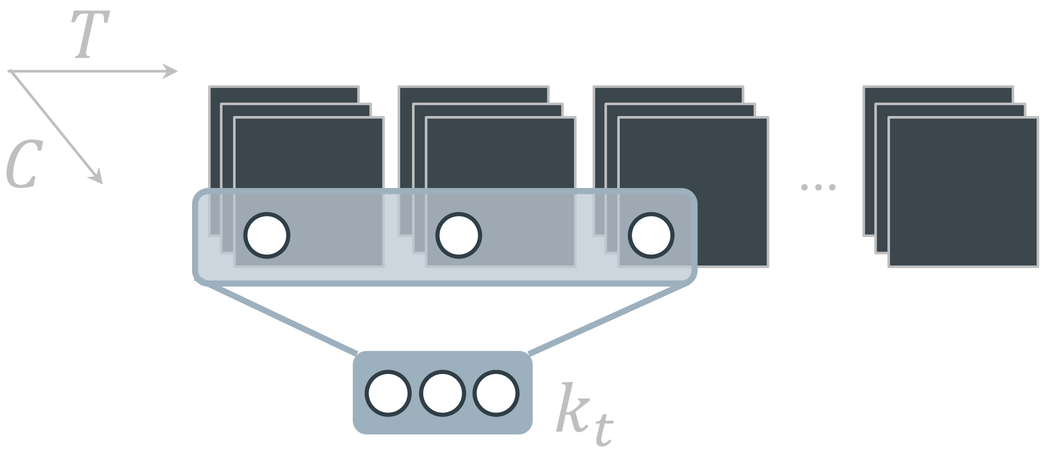
Model Overview

Timeception Layer

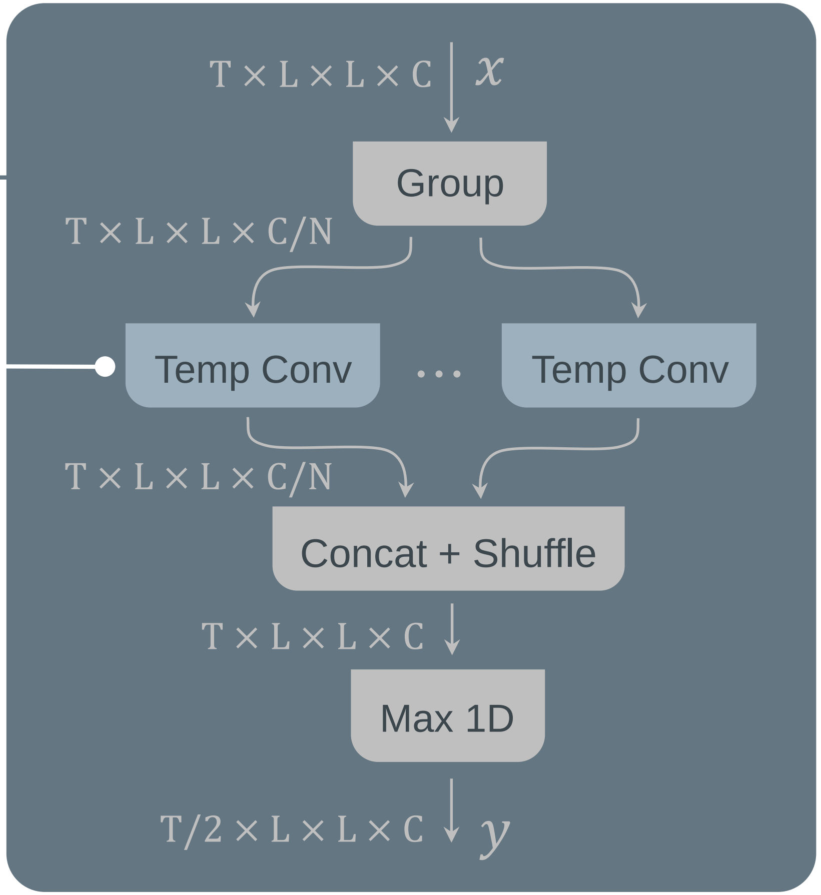
Method Efficiency

Grouped Conv

Depth-wise Conv



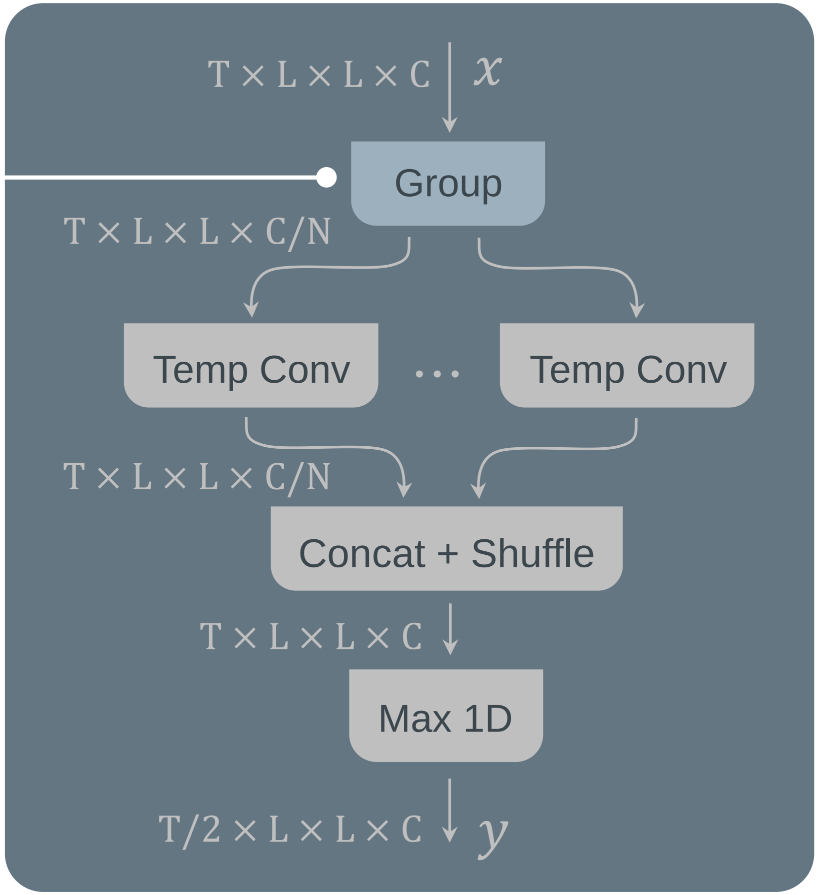
$$O(t \times c \times c) \rightarrow O(t \times c)$$


$\mathrm{T \times L \times L \times C}$
x
Group
$\mathrm{T \times L \times L \times C/N}$
Temp Conv
...
Temp Conv
$\mathrm{T \times L \times L \times C/N}$
Concat + Shuffle
$\mathrm{T \times L \times L \times C}$
Max 1D
$\mathrm{T/2 \times L \times L \times C}$
y

Timeception Layer

Method Efficiency

Grouped Conv


$T \times L \times L \times C$
x
Group
$T \times L \times L \times C/N$
Temp Conv
...
Temp Conv
$T \times L \times L \times C/N$
Concat + Shuffle
$T \times L \times L \times C$
Max 1D
$T/2 \times L \times L \times C$
y

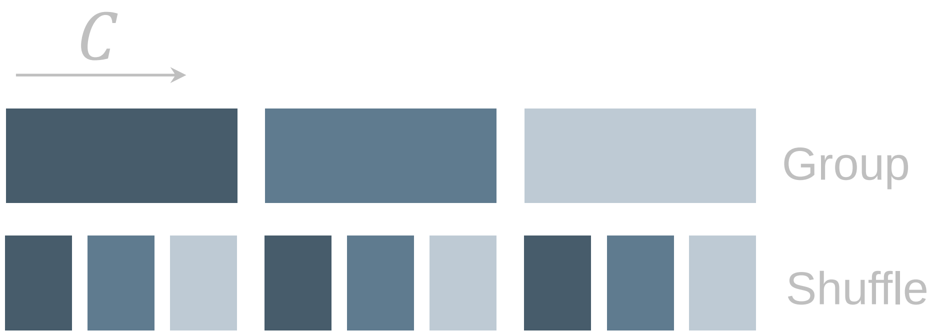
Timeception Layer

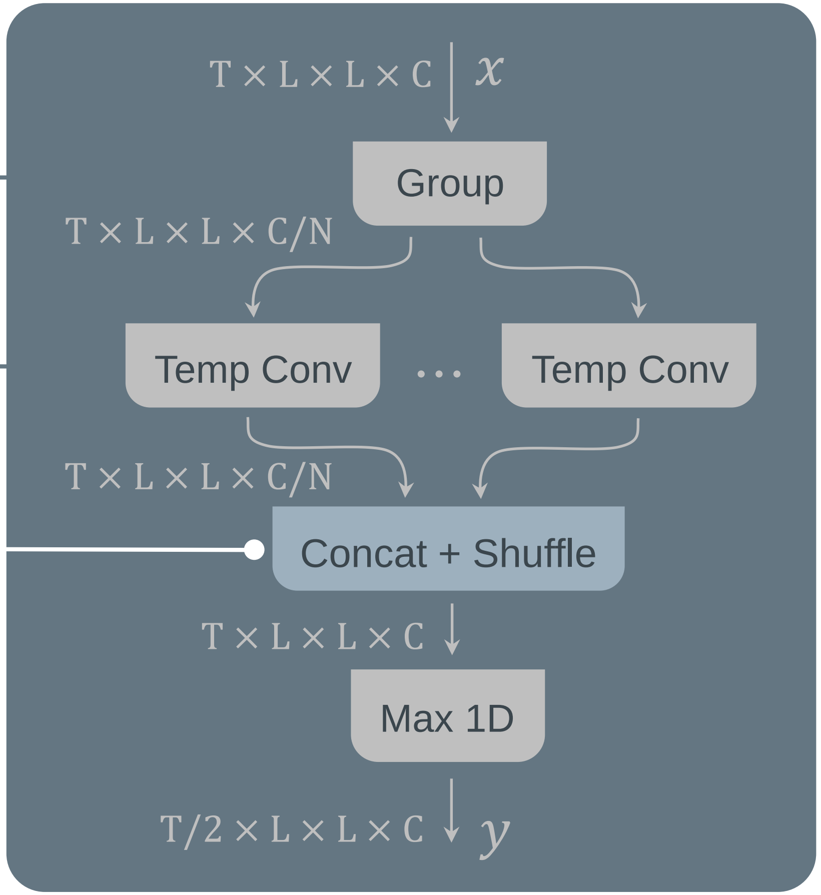
Method Efficiency

Grouped Conv

Depth-wise Conv

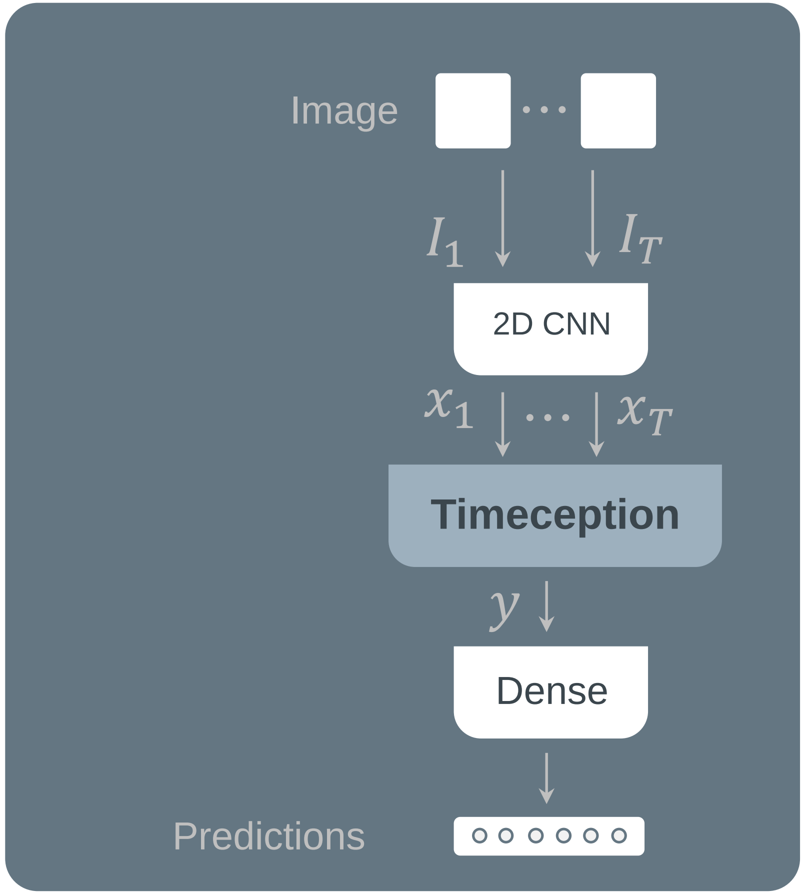
Channel Shuffle



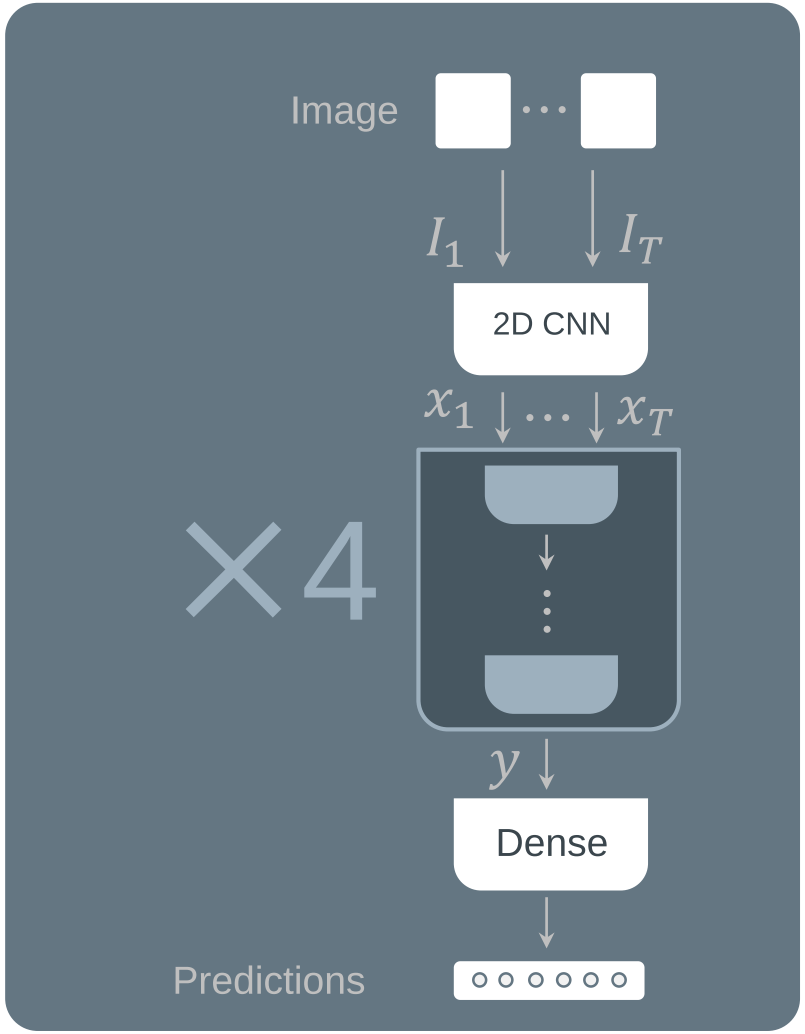

$T \times L \times L \times C$
x
Group
$T \times L \times L \times C/N$
Temp Conv
...
Temp Conv
$T \times L \times L \times C/N$
Concat + Shuffle
$T \times L \times L \times C$
Max 1D
$T/2 \times L \times L \times C$
y

Timeception Layer

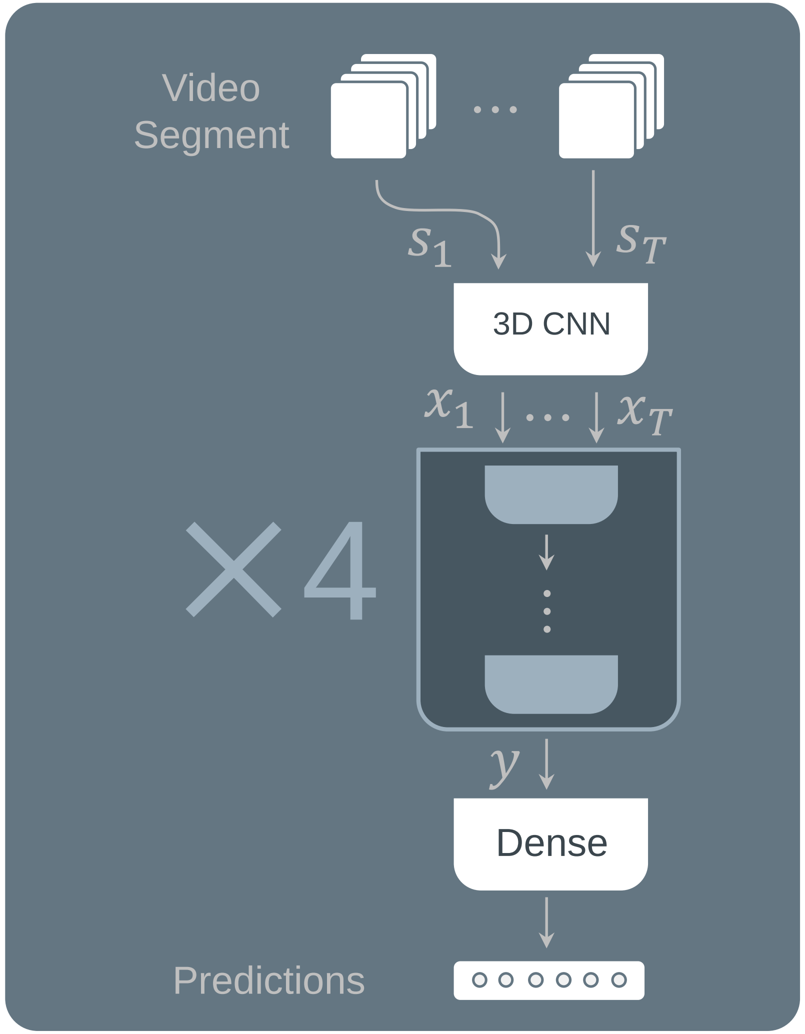
Method Timeception



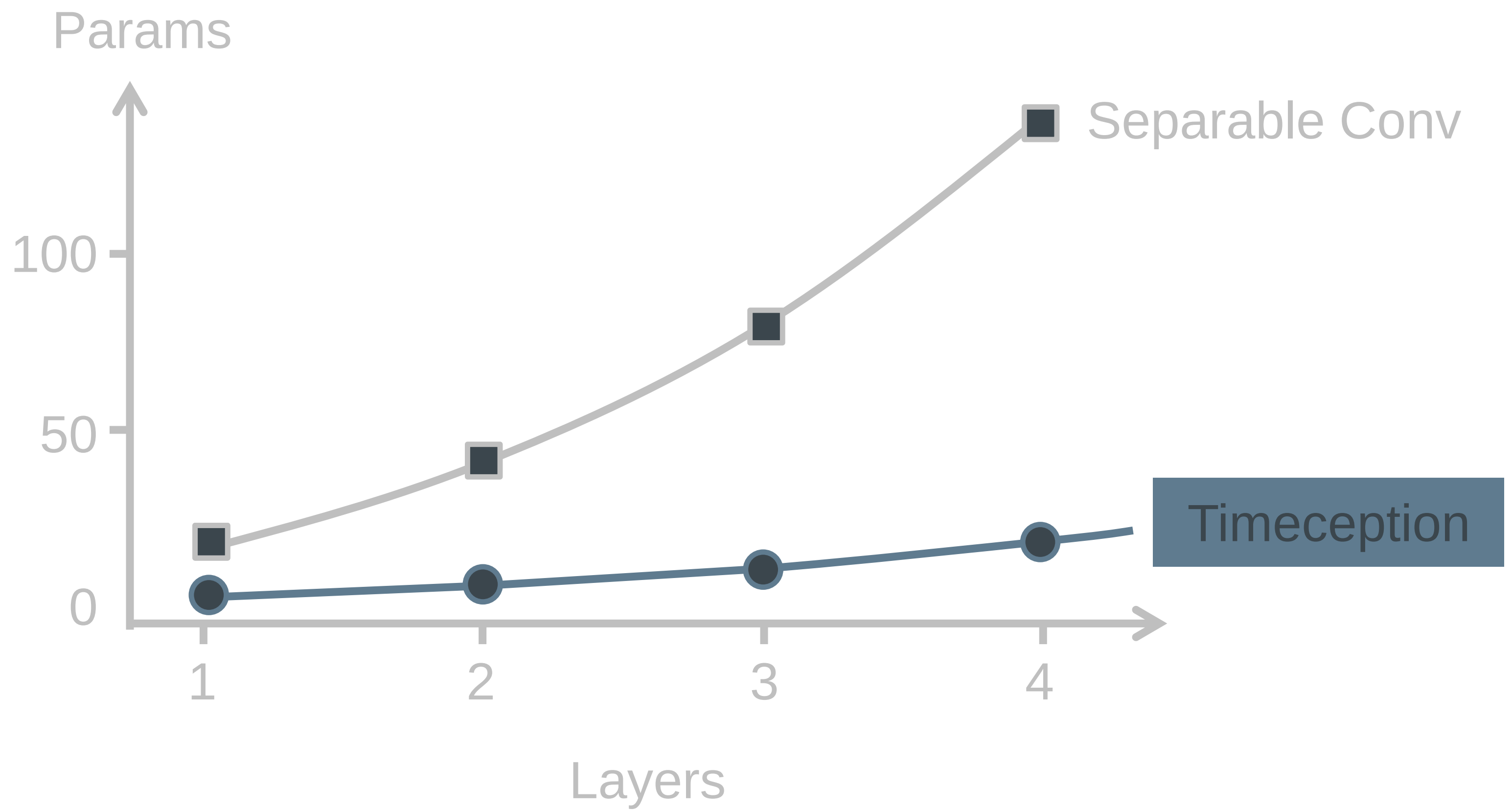
Method Timeception



Method Timeception

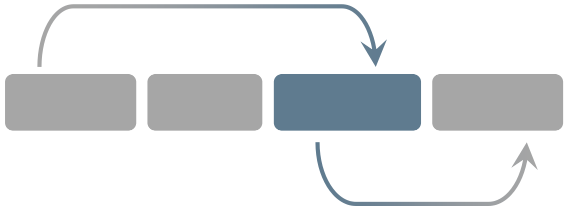


Method Efficiency

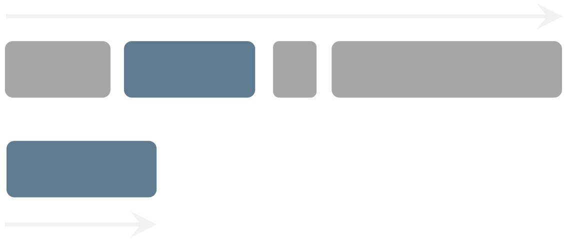


Method Timeception

1. Dependency

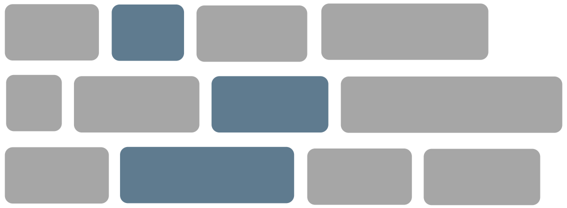


2. Long-range

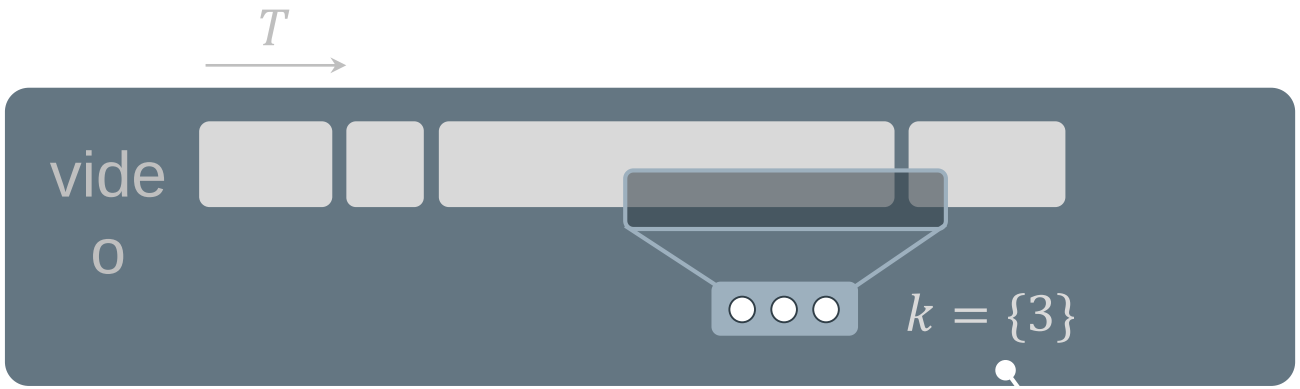


✓

3. Temporal Extent



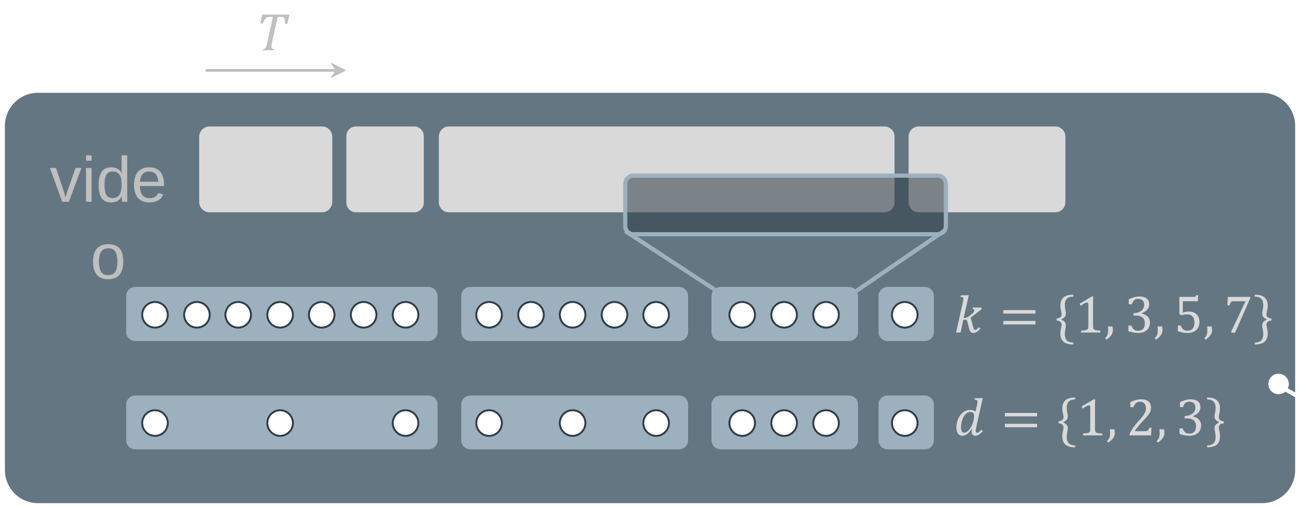
Method Tolerating Temporal Extents



Temporal Convolution

Fixed-size Kernel

Method Tolerating Temporal Extents



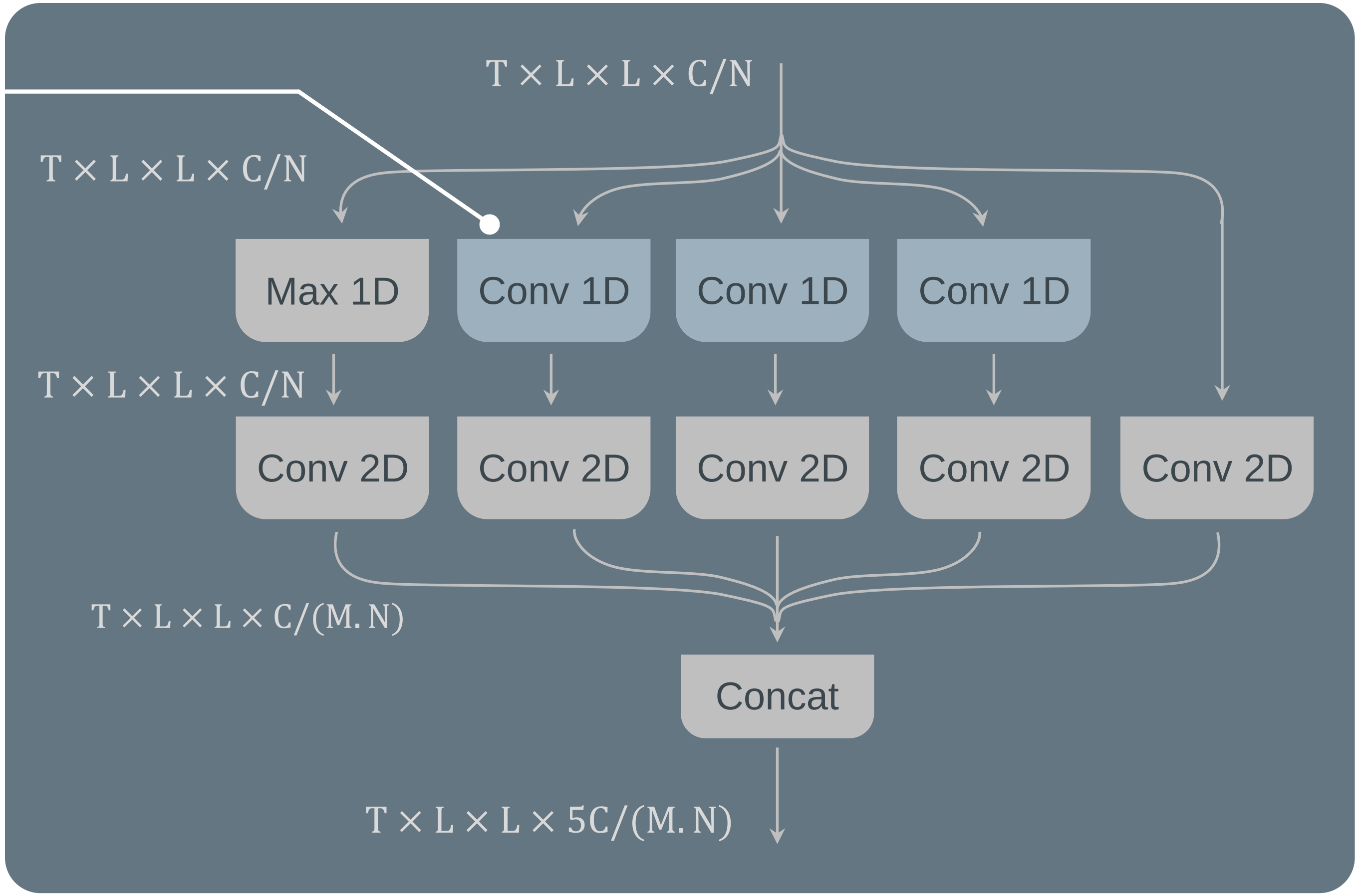
Temporal Convolution

Multi-scale Kernels

Method Tolerating Temporal Extents

Depthwise Temporal Conv

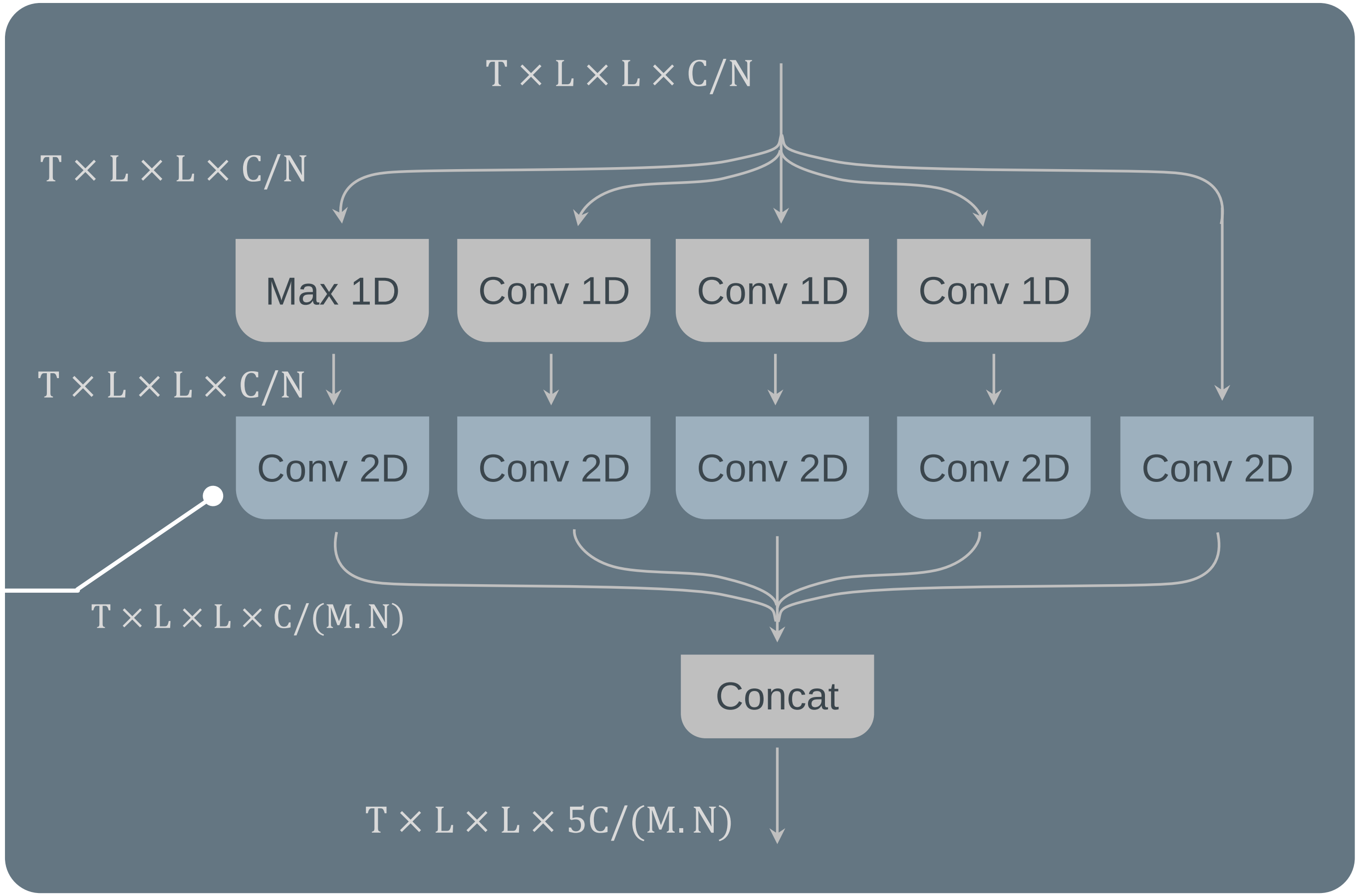
$k = \{3, 5, 7\}$



Temporal Conv Module

Method Tolerating Temporal Extents

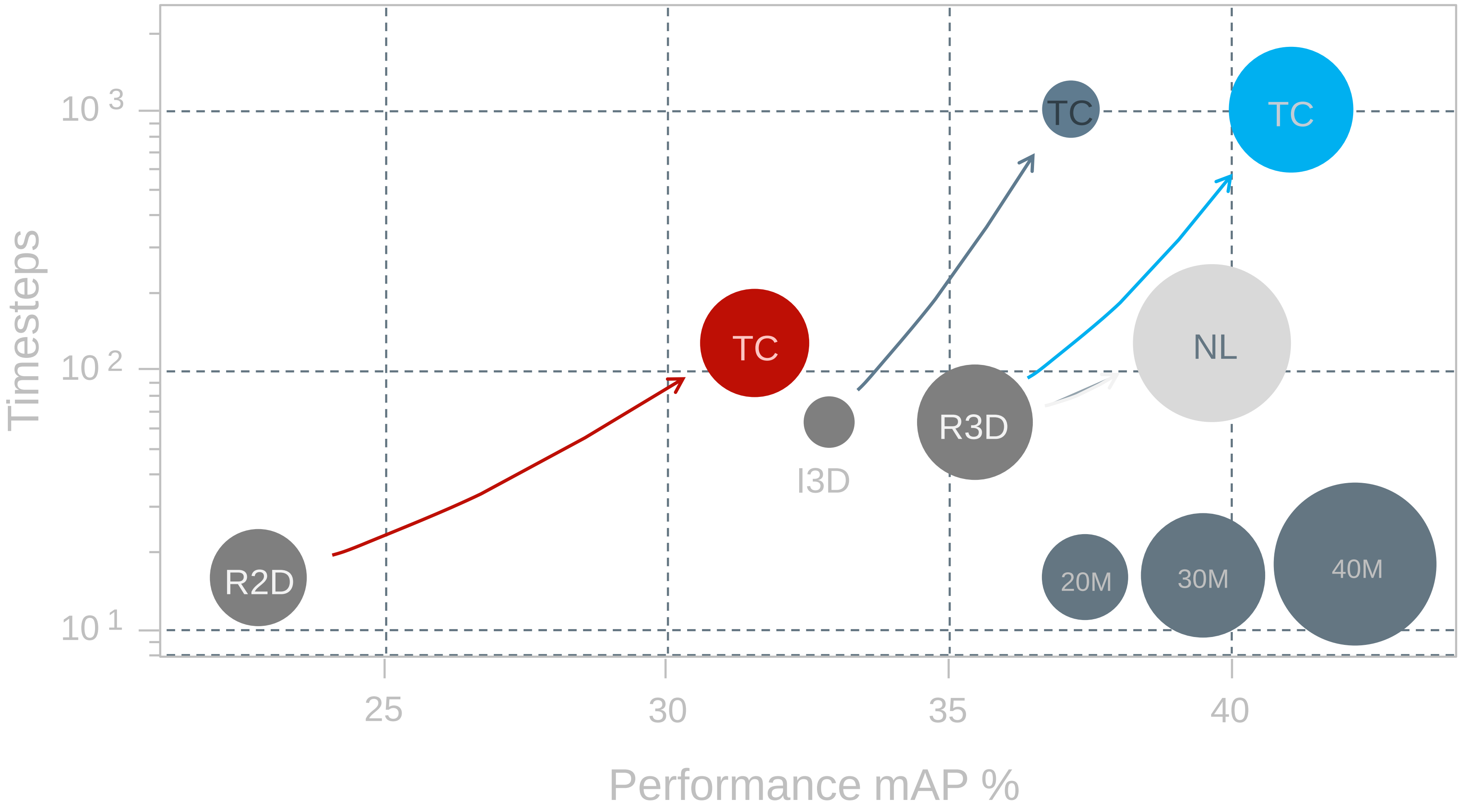
Channel Conv (1 × 1)



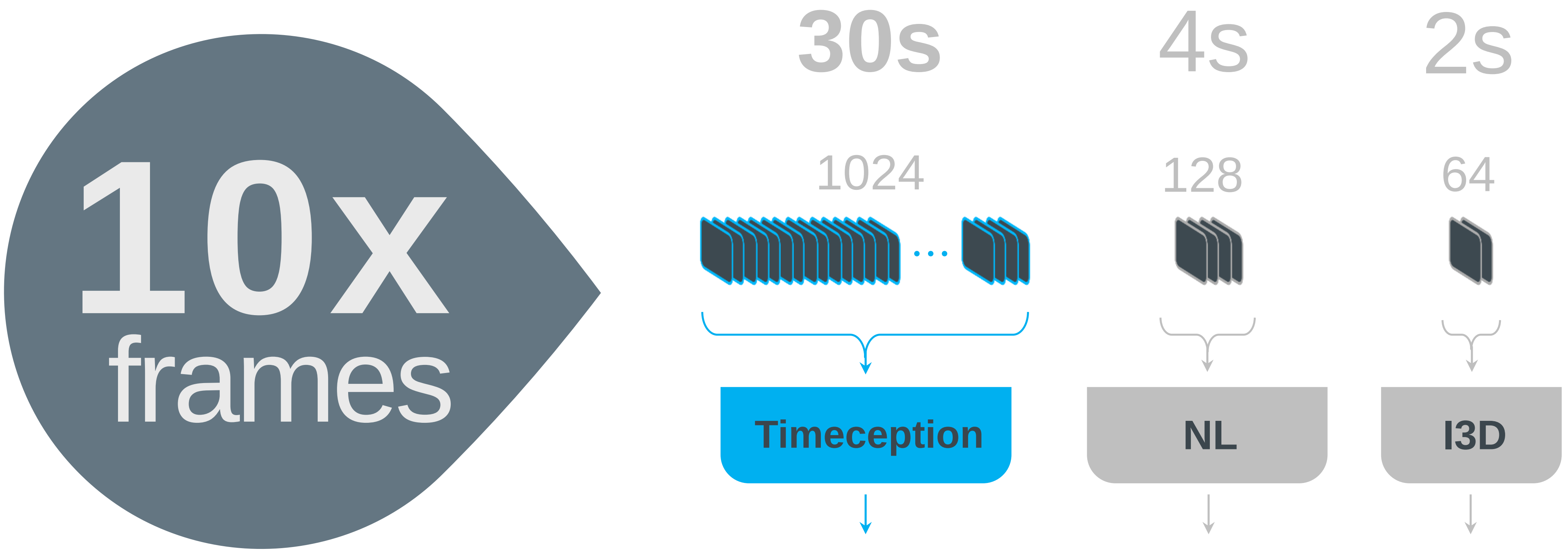
Temporal Conv Module

RESULTS

Results Charades Dataset



Results Temporal Footprint



10x frames

30s

1024

Timeception

4s

128

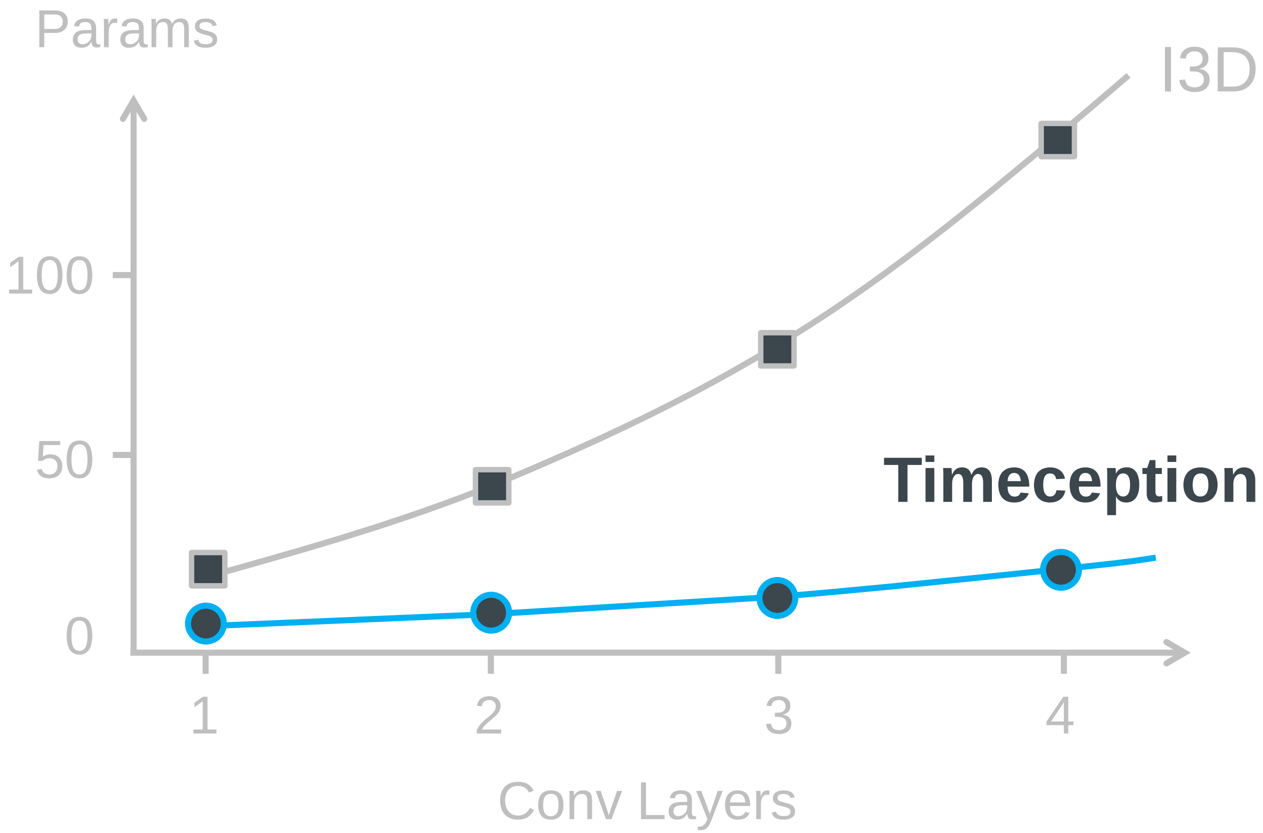
NL

2s

64

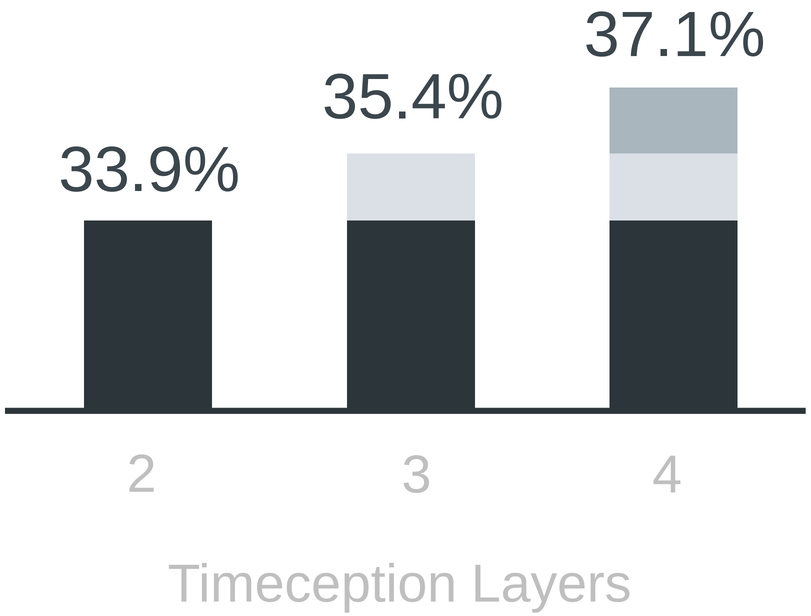
I3D

Results Layer Efficiency

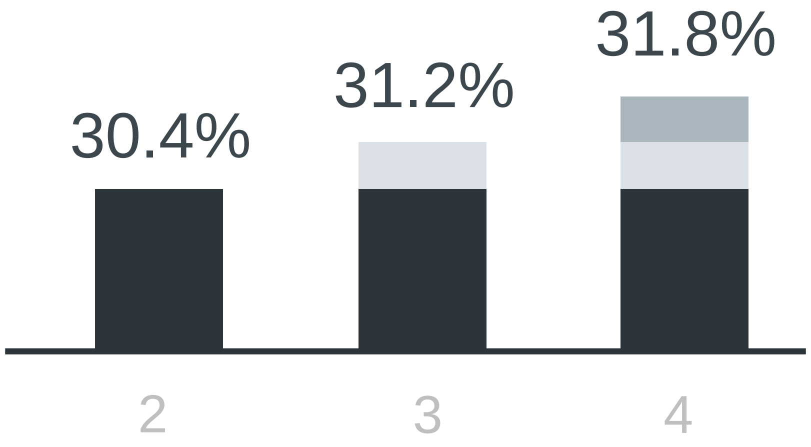


2.8m params

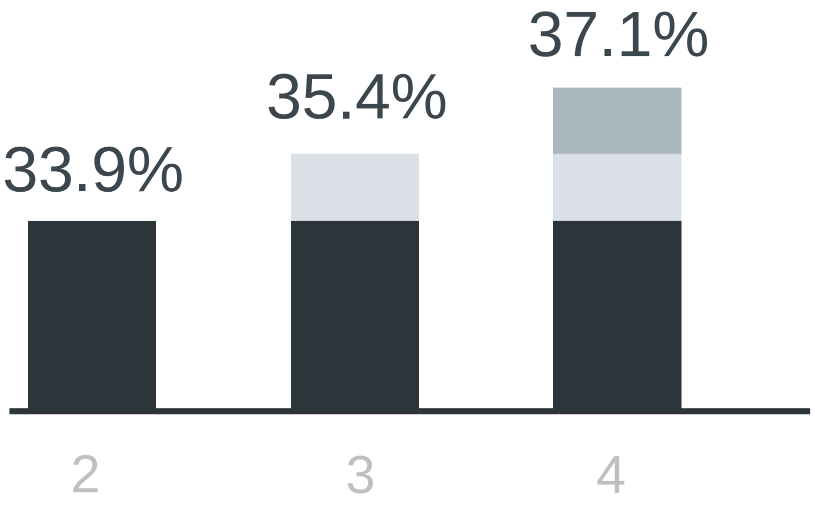
Results Layer Effectiveness



Results Layer Effectiveness

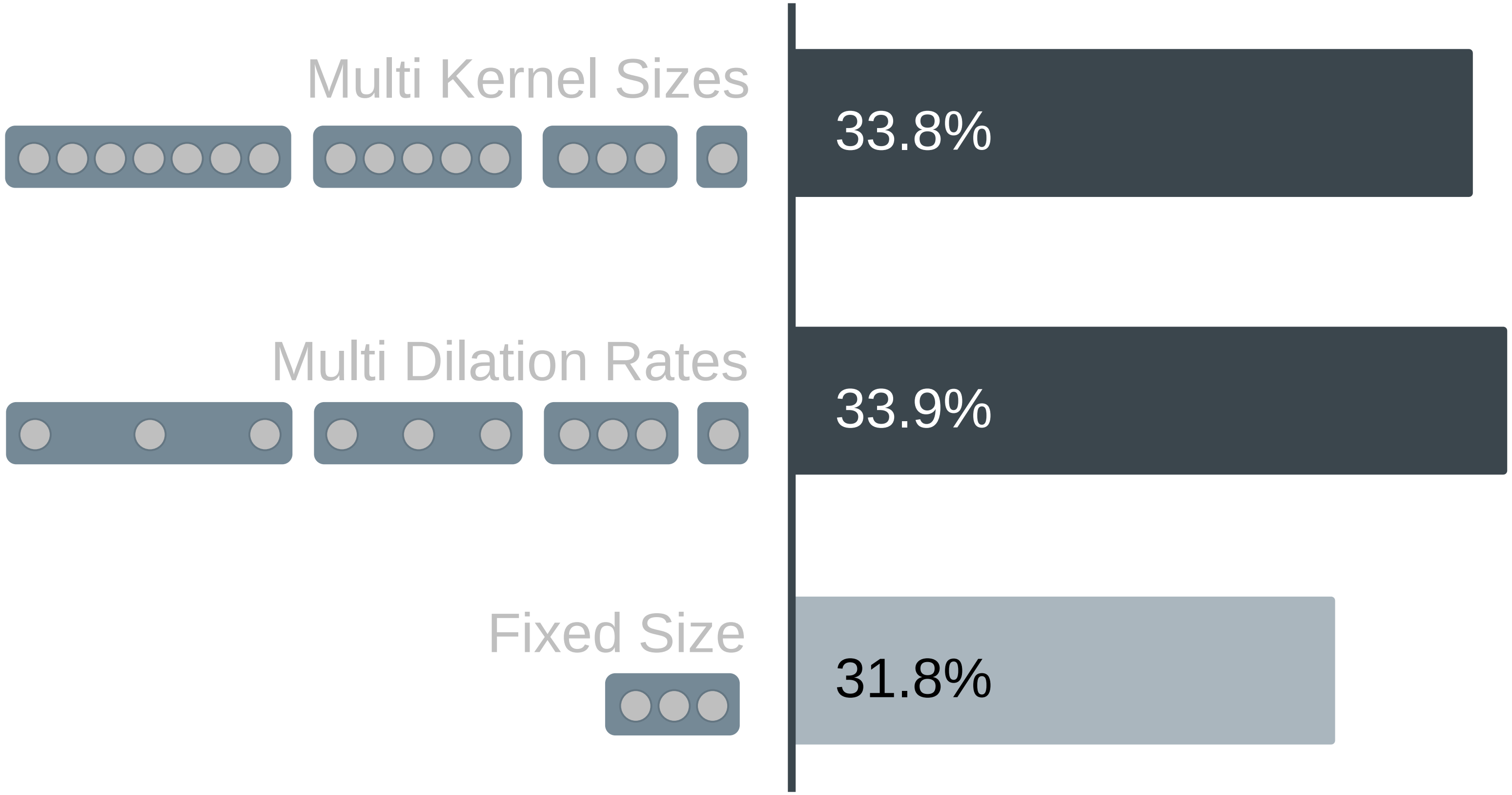


ResNet + Timeception



I3D + Timeception

Results Multi-Scale Kernels

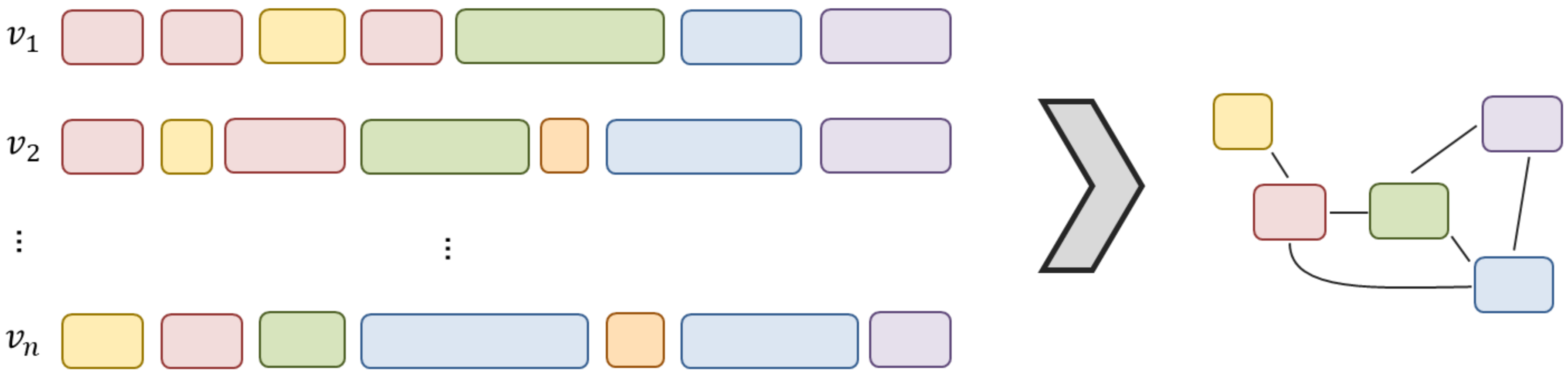


TIMECEPTION

N. Hussein, E. Gavves, A. Smeulders

Tuesday,
Oral 09.05
Poster 110

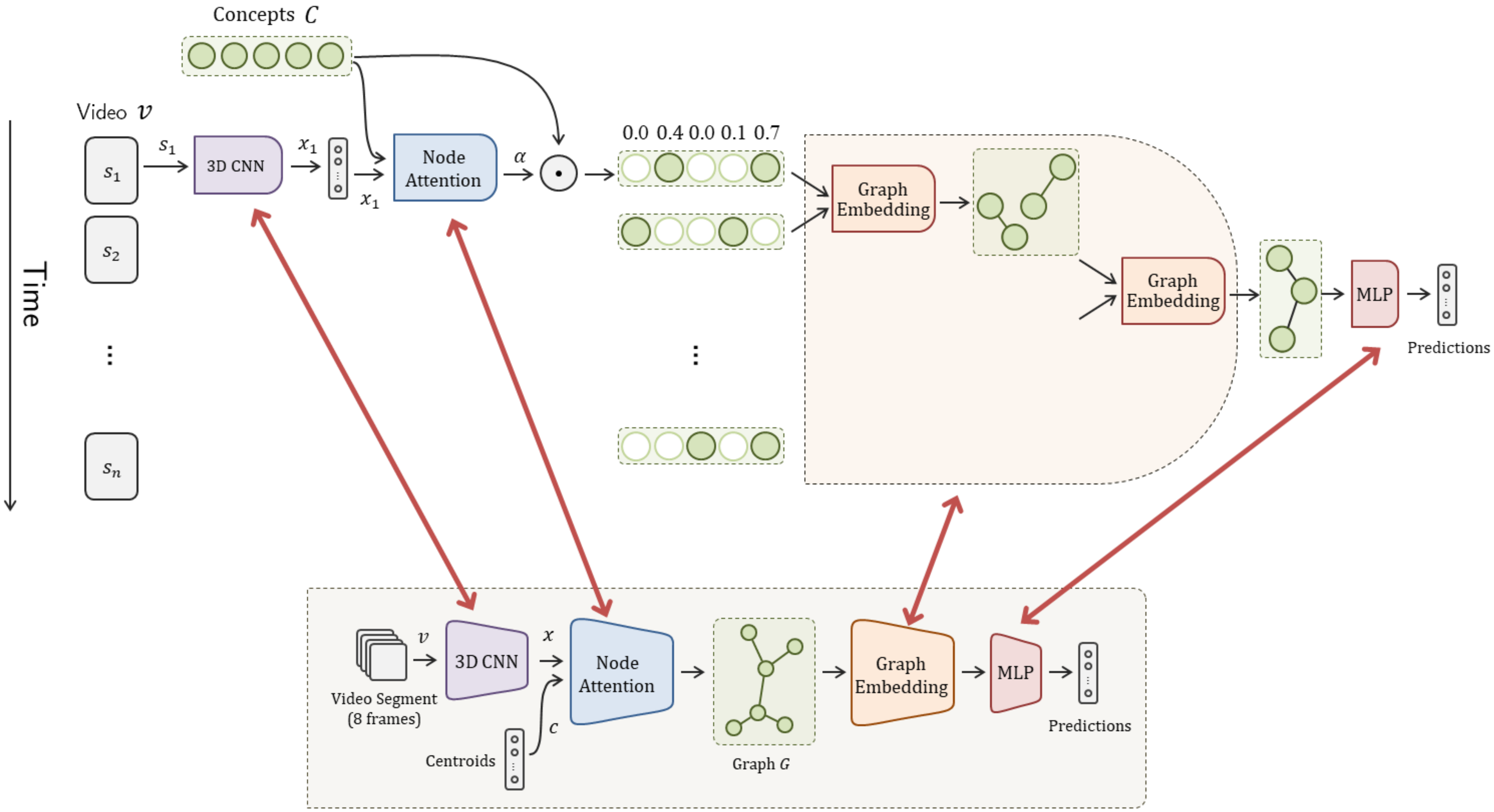
PUSHING THIS TO THE LIMIT: VIDEOGRAPH



Video Examples of “Preparing Coffee”

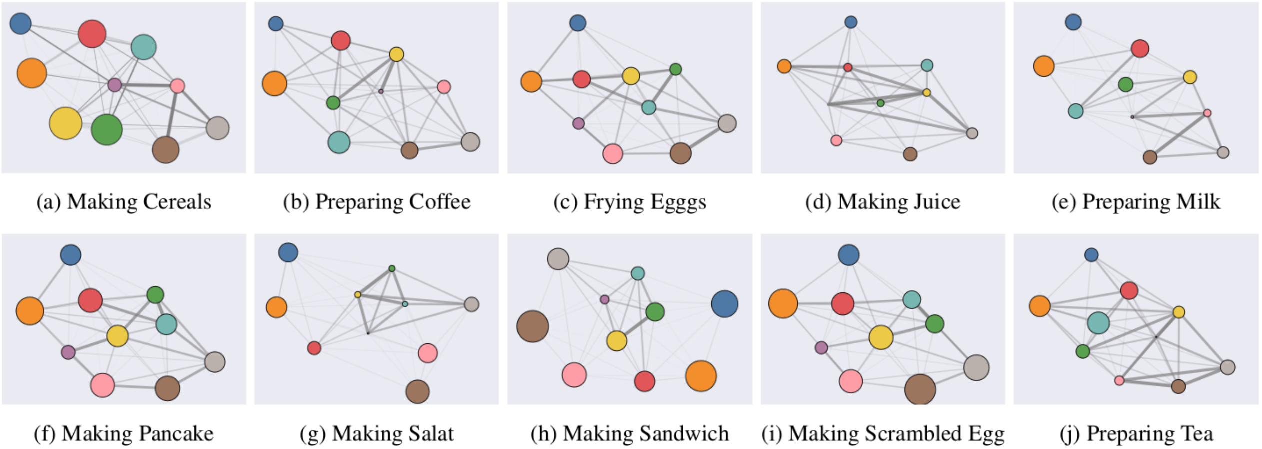
Graph-based Representation

VIDEOGRAPH

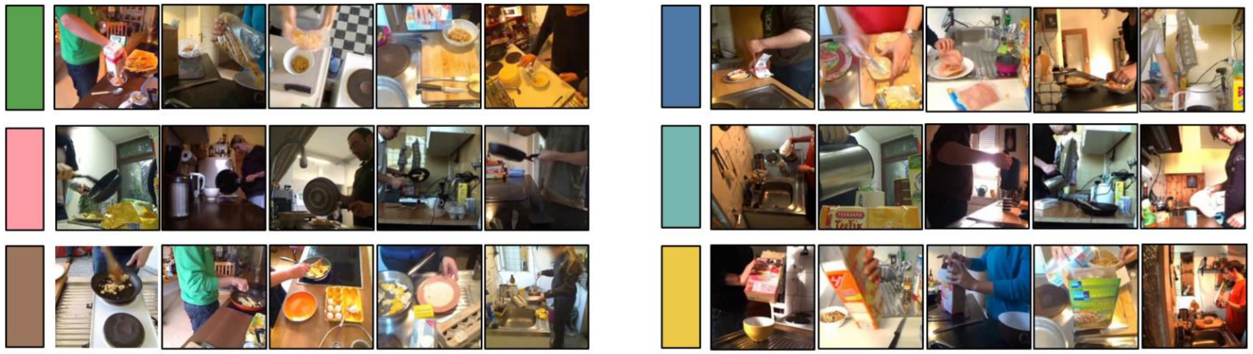

Concepts C
Video v
Time
s_1
s_2
s_n
s_1
3D CNN
x_1
x_1
Node Attention
α
⊙
0.0 0.4 0.0 0.1 0.7
Graph Embedding
Graph Embedding
MLP
Predictions
Video Segment (8 frames)
v
3D CNN
x
c
Centroids
Node Attention
Graph G
Graph Embedding
MLP
Predictions

EXPERIMENTS

Method	Modality	mAP (%)
Two-stream [17]	RGB + Flow	18.6
Two-stream + LSTM [17]	RGB + Flow	17.8
ActionVLAD [5]	RGB + iDT	21.0
Temporal Fields [17]	RGB + Flow	22.4
Temporal Relations [23]	RGB	25.2
ResNet-152 [61]	RGB	22.8
ResNet-152 + Timeception [2]	RGB	31.6
I3D [9]	RGB	32.9
I3D + ActionVLAD [5]	RGB	35.4
I3D + Timeception [2]	RGB	37.2
I3D + VideoGraph	RGB	**37.8**


(a) Making Cereals (b) Preparing Coffee (c) Frying Egggs (d) Making Juice (e) Preparing Milk
(f) Making Pancake (g) Making Salat (h) Making Sandwich (i) Making Scrambled Egg (j) Preparing Tea

● cereal, ● pan, ● eggs, ● sandwitch, ● kettle, and ● foodbox



TIMECEPTION/VIDEOGRAPH: WHAT HAVE WE LEARNED?

- Scaling up in time is possible if you do smart decomposition of the operations

- Larger models don't have to mean immense parameters or computation times

- Organizing learned representations in graphs allows for clustering visual concepts reliably
  - Explainable action recognition ?

TIMECEPTION: OPEN QUESTIONS

Can we go larger? Movie-long video?
Action detection in long videos?
Infinite long videos → Streaming?
Integrate dynamics learning more explicitly for fine grained complex actions?
Natively efficient video models?

THANK YOU!