Deep Structured Models for Human Activity Recognition

Greg Mori
School of Computing Science
Simon Fraser University

What does activity recognition involve?


Detection: are there people?

Objects and scenes: where are they?

Action recognition: what are they doing?

Intention/social role: why are they doing this?

Group activity recognition: what is the overall situation?

These are inter-related problems:
model structures

Desiderata for Activity Recognition Models

Label structure

Hu et al., CVPR 16
Deng et al., CVPR 16
Nauata et al., CVPRW 17
Deng et al., CVPR 17

Temporal structure

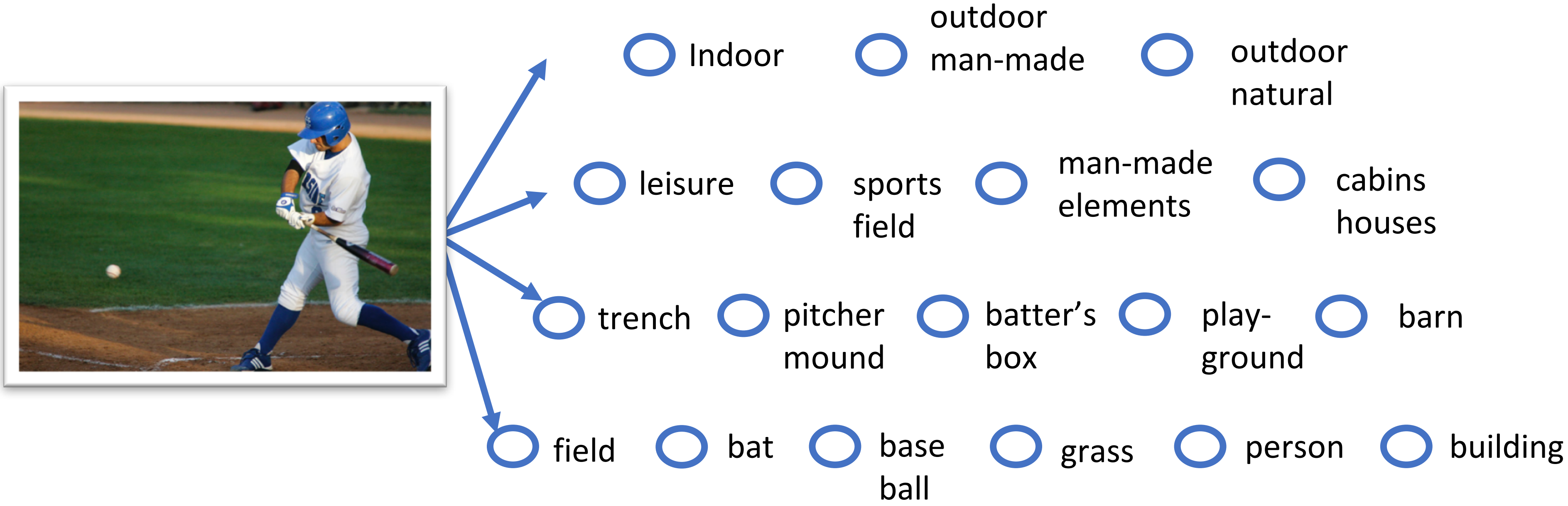
Yeung et al., CVPR 16
Yeung et al., IJCV 17
He et al., WACV 18
Chen et al., ICCVW 17

Group structure

Ibrahim et al., CVPR 16
Mehrasa et al., arXiv 17
Khodabandeh et al., arXiv 17
Lan et al. CVPR 12

Image Classification

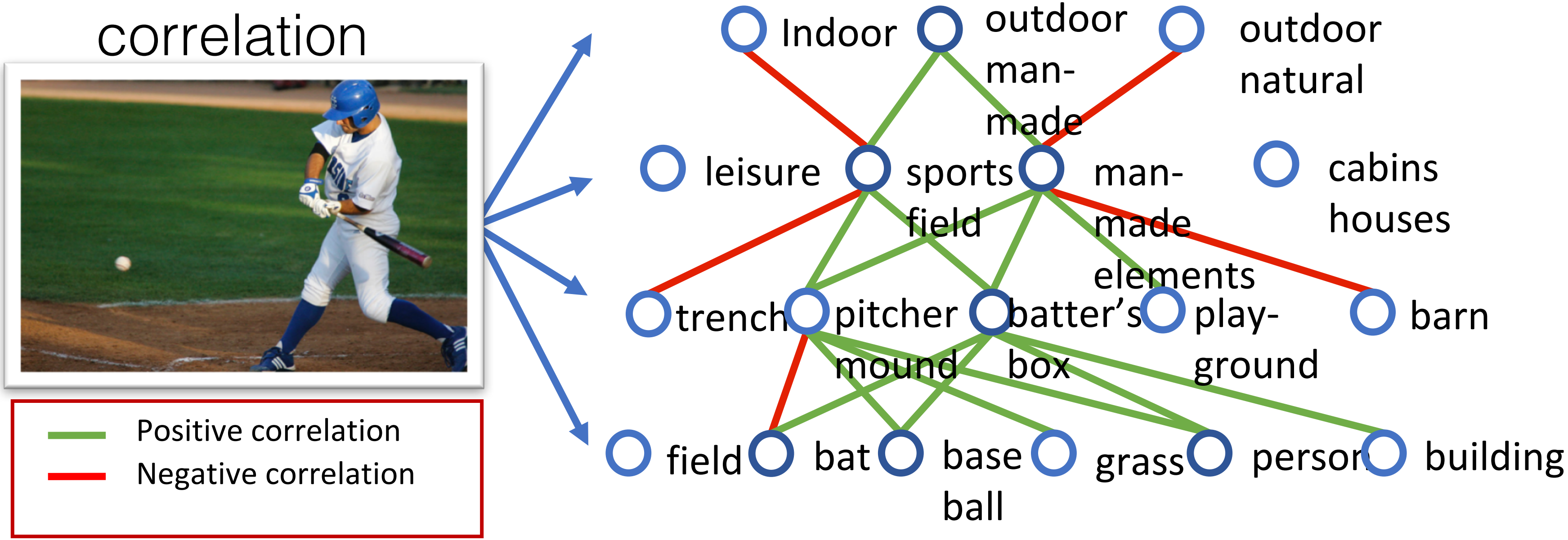
- A natural image can be categorized with labels at different concept layers



Hu, Deng, Zhou, Liao, Learning Structured Inference Neural Networks with Label Relations, CVPR 2016

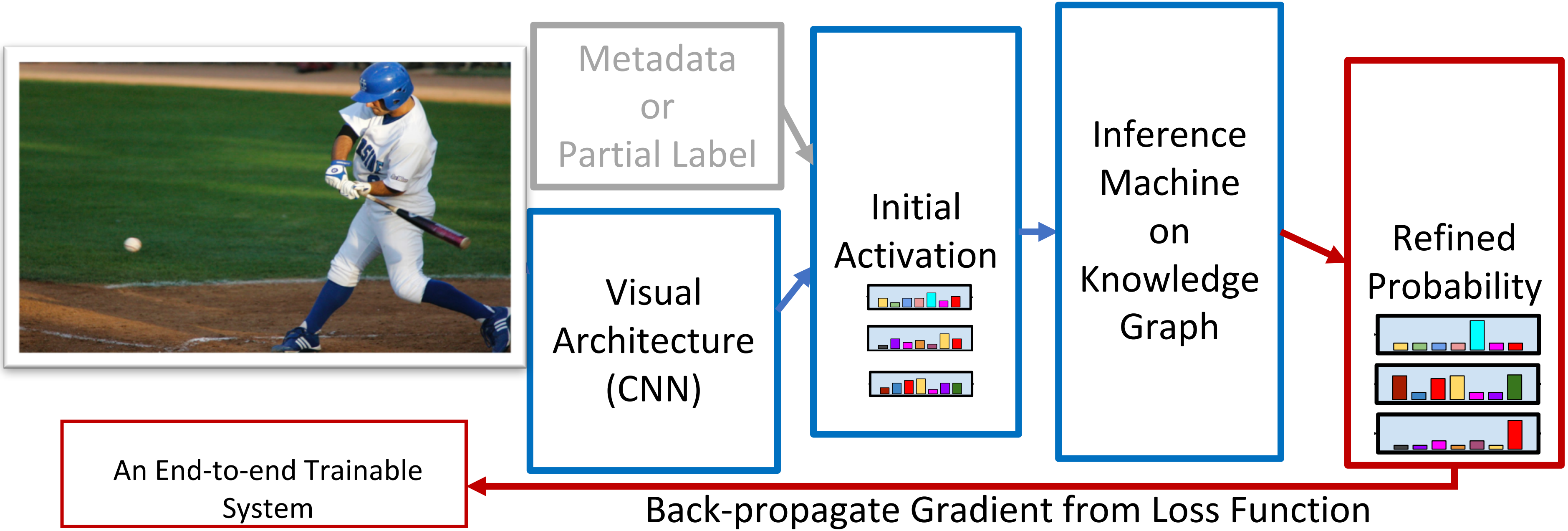
Label Correlation Helps

- Such categorization at different concept layers can be modeled with label graphs
- It is natural and straightforward to leverage label correlation



Goal: A generic label relation model

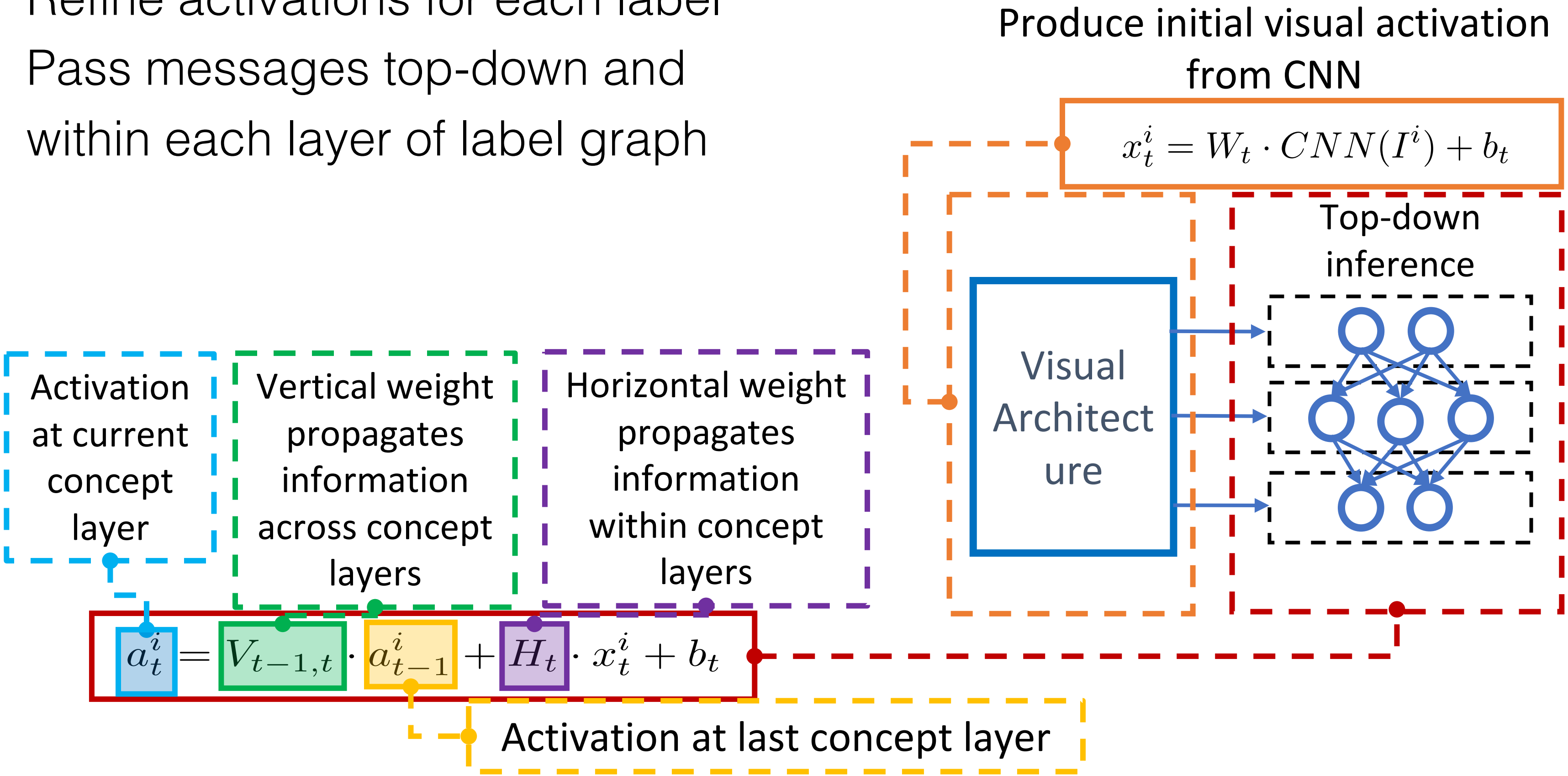
- Infer the entire label space from visual input
- Infer missing labels given a few fixed provided labels



Hu, Deng, Zhou, Liao, Learning Structured Inference Neural Networks with Label Relations, CVPR 2016

Top-down Inference Neural Network

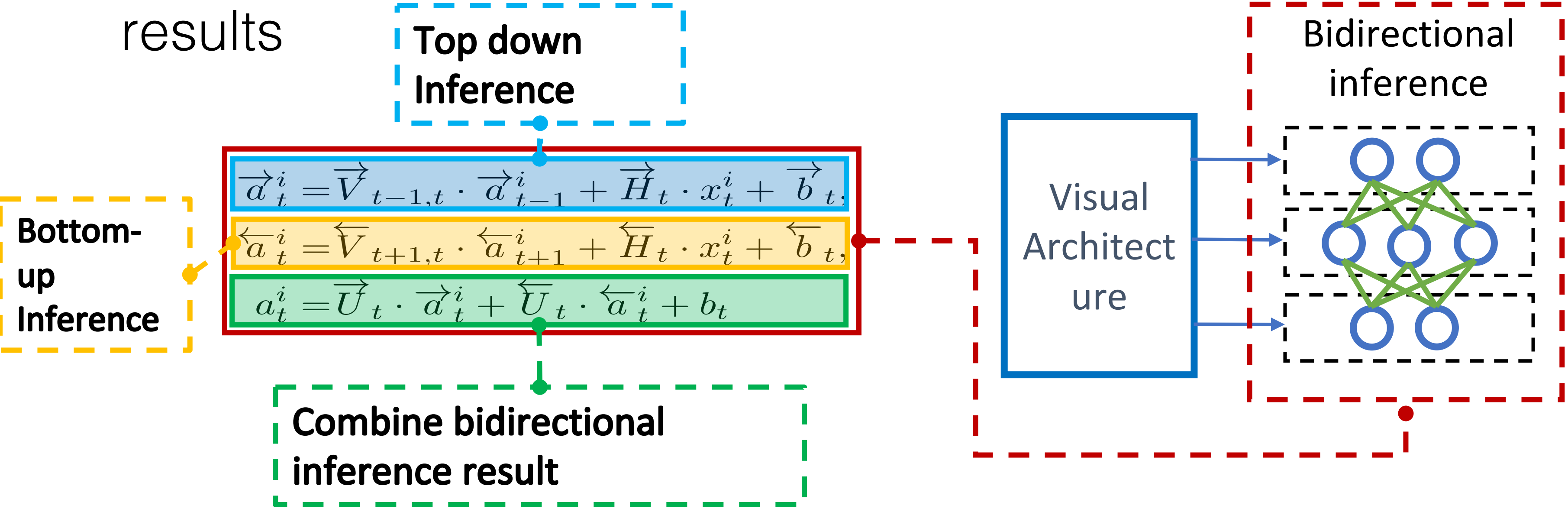
- Refine activations for each label
- Pass messages top-down and within each layer of label graph



Hu, Deng, Zhou, Liao, Learning Structured Inference Neural Networks with Label Relations, CVPR 2016

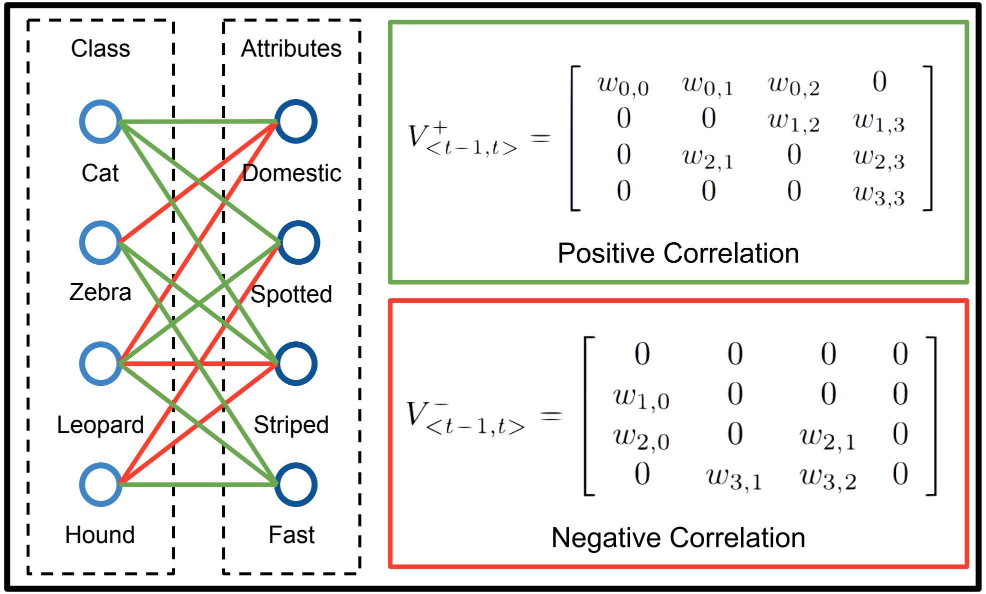
Bidirectional Inference Neural Network (BINN)

- Bidirectional inference to make information propagate across entire label structure
- Inference in each direction independently and blend results



Structured Inference Neural Network (SINN)

- BINN is hard to train
- Regularize connections with prior knowledge about label correlations
- Decompose connections into Positive correlation + Negative correlation



Hu, Deng, Zhou, Liao, Learning Structured Inference Neural Networks with Label Relations, CVPR 2016

Structured Inference Neural Network (SINN)

- Evolve BINN formulation with regularization in connections

Negative Component

Positive Component

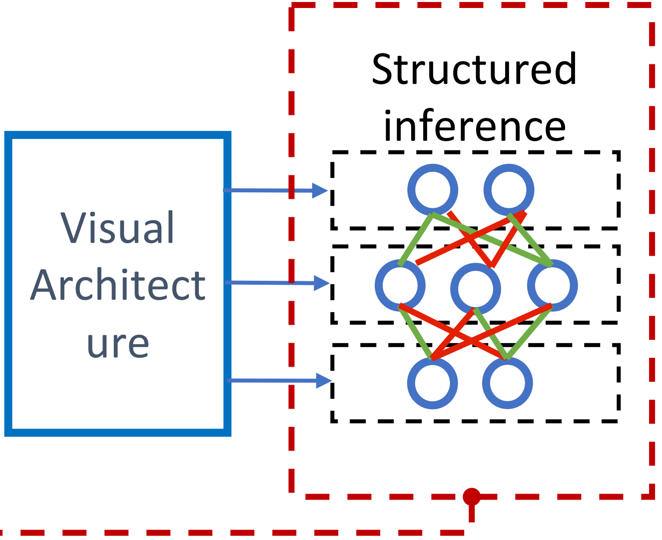
$$\overrightarrow{a}_t^i = \gamma(\overrightarrow{V}^+_{t-1,t} \cdot \overrightarrow{a}^i_{t-1}) + \gamma(\overrightarrow{H}^+_t \cdot x_t^i) - \gamma(\overrightarrow{V}^-_{t-1,t} \cdot \overrightarrow{a}^i_{t-1}) - \gamma(\overrightarrow{H}^-_t \cdot x_t^i) + \overrightarrow{b}_t,$$

$$\overleftarrow{a}_t^i = \gamma(\overleftarrow{V}^+_{t+1,t} \cdot \overleftarrow{a}^i_{t+1}) + \gamma(\overleftarrow{H}^+_t \cdot x_t^i) - \gamma(\overleftarrow{V}^-_{t+1,t} \cdot \overleftarrow{a}^i_{t+1}) - \gamma(\overleftarrow{H}^-_t \cdot x_t^i) + \overleftarrow{b}_t,$$

$$a_t^i = \overrightarrow{U}_t \cdot \overrightarrow{a}_t^i + \overleftarrow{U}_t \cdot \overleftarrow{a}_t^i + b_t$$

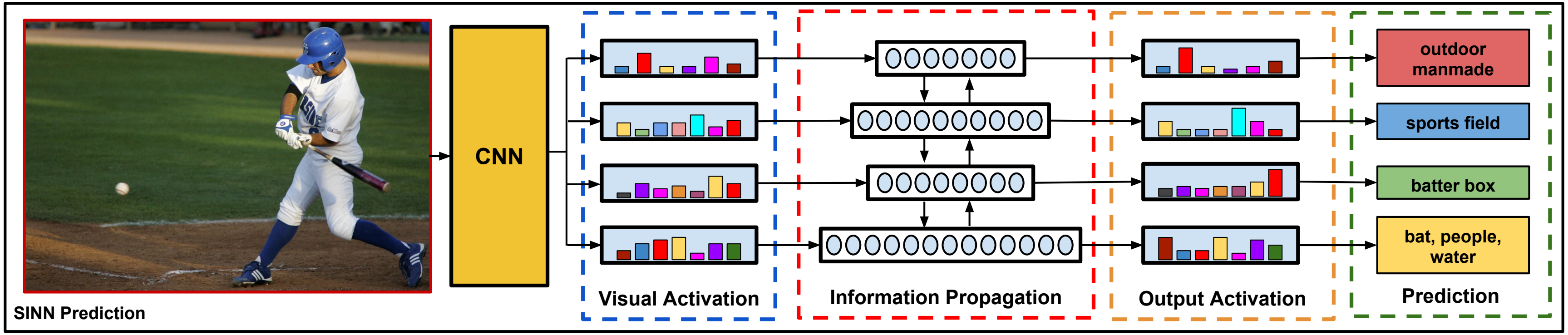
ReLU neuron is **essential** to keep positive/negative contribution

$\gamma(x) = ReLU(x)$



Prediction from Purely Visual Input

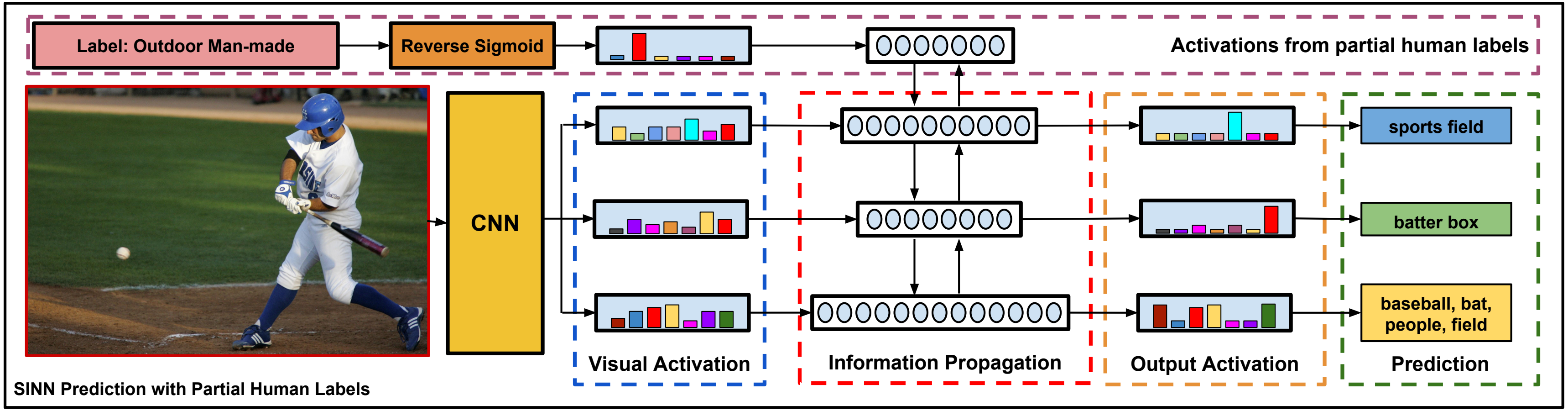
- Visual architecture (e.g. Convolutional Neural Network) produces visual activation
- SINN implements information propagation bidirectionally and produces refined output activation



Hu, Deng, Zhou, Liao, Learning Structured Inference Neural Networks with Label Relations, CVPR 2016

Prediction with Partially Observed Labels

- Reverse Sigmoid (logit) neuron produces activation from Partial labels
- SINN adapts both visual activation and activation from partial labels to infer the remaining labels



Reverse sigmoid (logit): produce activation from label

- Reverse the sigmoid function to produce sigmoid input

Inverse of sigmoid

$$y = \sigma(x) = \frac{1}{1 + \exp^{-x}}$$

$$a(y) = \log \frac{1}{1 - g(y)},$$

$$g(y) = \begin{cases} y + \epsilon, & \text{if } y = 0, \\ y - \epsilon, & \text{if } y = 1. \end{cases}$$

Use a small **epsilon** to keep numerical stability (0.005)

Image Datasets

- Evaluate with two types of experiments on three datasets

Animals with Attributes
[Lampert et al. 2009]

Labels
28 taxonomy terms
50 animal classes
85 attributes

Task: predict entire label set

- Taxonomy terms are constructed from Word Net as [Hwang et al. 2012]
- Knowledge graph constructed by combining class-attributes graph with taxonomy graph

NUS-WIDE
[Chua et al. 2009]

Labels
698 image groups
81 concepts
1000 tags

Task: predict 81 concepts with observing tags/image groups

- Knowledge graph produced by Word Net using semantic similarity
- 698 image groups constructed from image meta data

SUN 397
[Xiao et al. 2012]

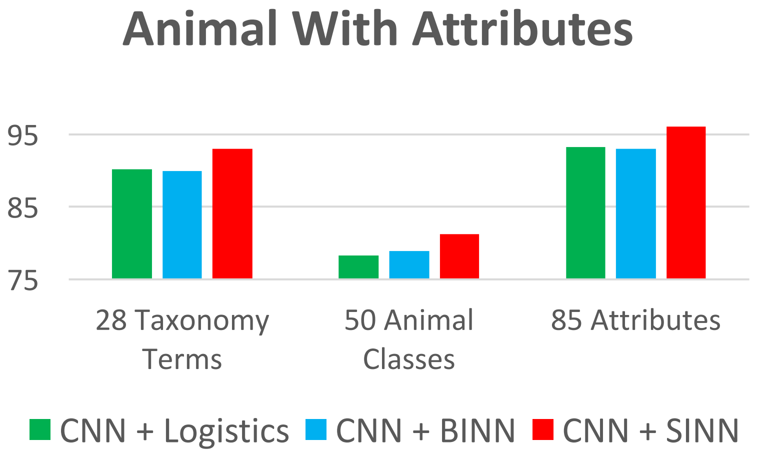
Labels
3 coarse
16 general
397 fine-grained

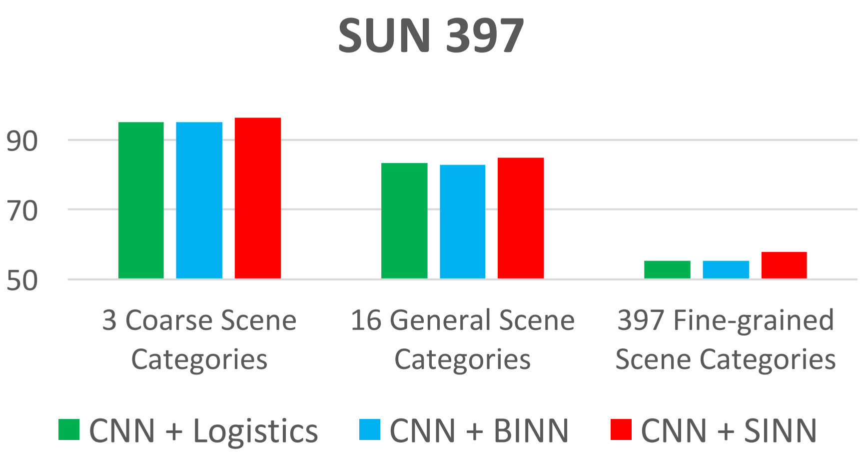
Task 1: predict entire label set
Task 2: predict fine-grained scene given coarse scene category

- Knowledge graph provided by dataset

Ex1: Inference from visual input

- Produce predictions on entire label space
- Evaluate on each concept layer (measured by mAP per class)
- Consistent improvement over baselines on different concept layers





Ex2: Inference from partial labels (NUS-WIDE)

- Produce predictions given partial 1k tags and 698 image groups

Ground Truth: railroad
CNN + Logistic: statue buildings person
Our Predictions: railroad person sky

Ground Truth: animal grass water dog
CNN + Logistic: grass person animal
Our Predictions: water animal dog

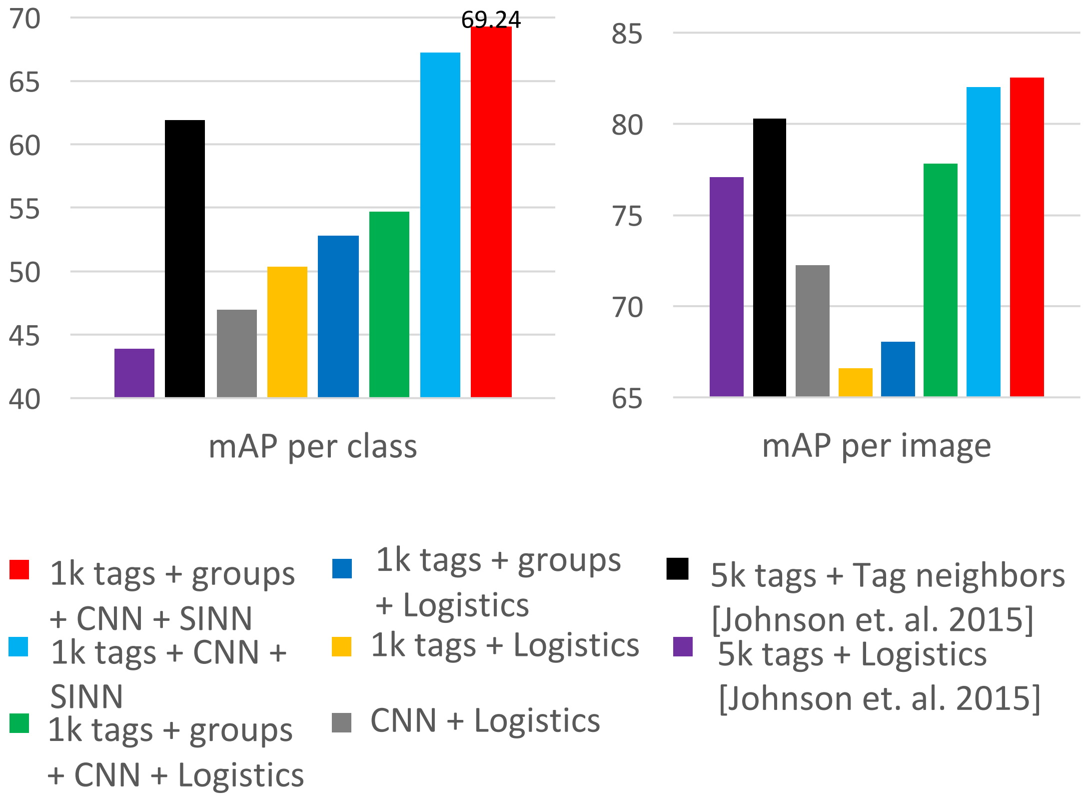
Ground Truth: rainbow clouds sky
CNN + Logistic: clouds water sky
Our Predictions: rainbow clouds sky

Ground Truth: food water
CNN + Logistic: food plants flower
Our Predictions: food plants water

Correct predictions are marked in **blue** while incorrect are marked in **red**

Ex2: Inference from partial labels (NUS-WIDE)

- Evaluate on standard 81 ground truth classes of NUSWIDE
- Outperform all baselines by large margin



Hu, Deng, Zhou, Liao, Learning Structured Inference Neural Networks with Label Relations, CVPR 2016

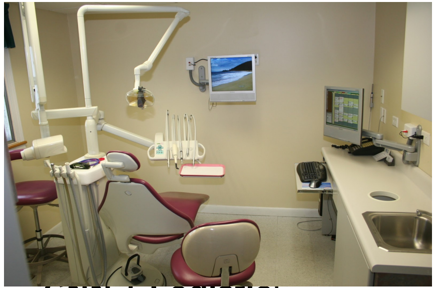
Ex2: Inference with partial labels (SUN397)

- Produce predictions given coarse-level labels (3 coarse categories)

CNN + Logistic: campus
Observed Label: outdoor/man-made
Our Predictions: abbey
Ground Truth: abbey

CNN + Logistic: building facade
Observed Label: outdoor/man-made
Our Predictions: library/outdoor
Ground Truth: library/outdoor

CNN + Logistic: patio
Observed Label: outdoor/natural; outdoor/man-made
Our Predictions: picnic area
Ground Truth: picnic area

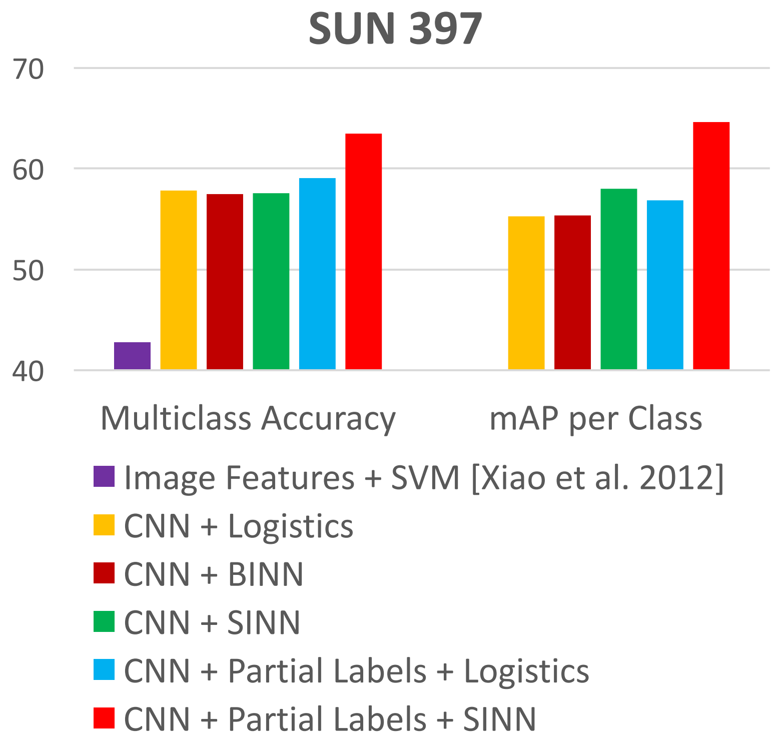


CNN + Logistic: operating room
Observed Label: indoor
Our Predictions: dentists office
Ground Truth: dentists office

Correct predictions are marked in **blue** while incorrect are marked in **red**

Ex2: Inference with partial labels (SUN397)

- Evaluate on 397 fine-grained scene categories
- Significantly improved performance

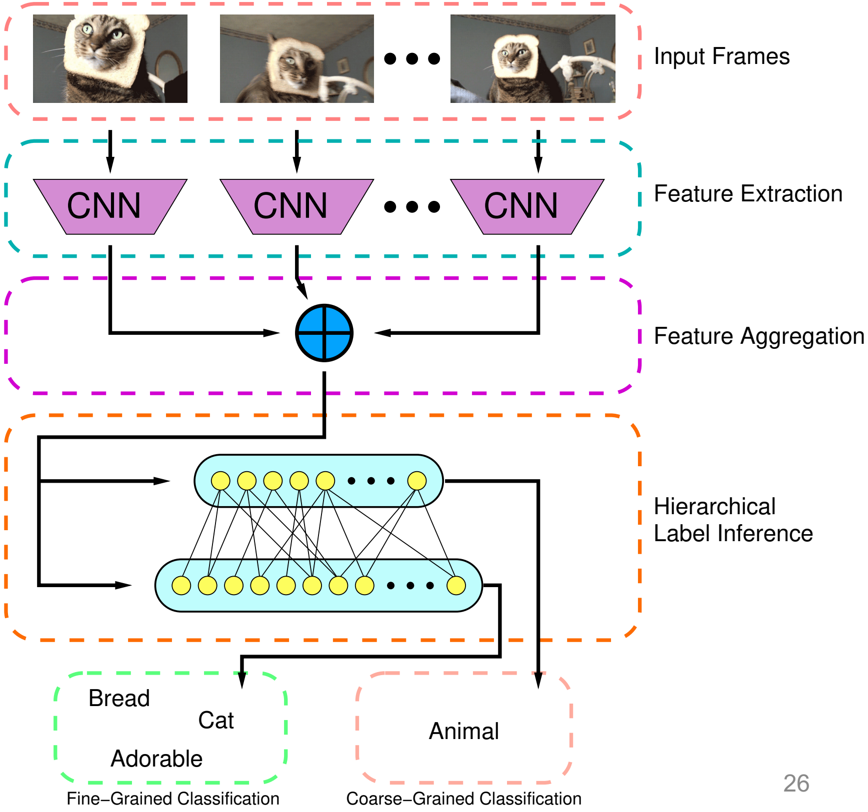


Hu, Deng, Zhou, Liao, Learning Structured Inference Neural Networks with Label Relations, CVPR 2016

Video Dataset: YouTube-8M

- Youtube-8M V1 / V2
  - 8 million / 7 million videos
  - ~500K hours of video
  - 4800 possible labels
  - 1.8 / 3.4 labels per video average

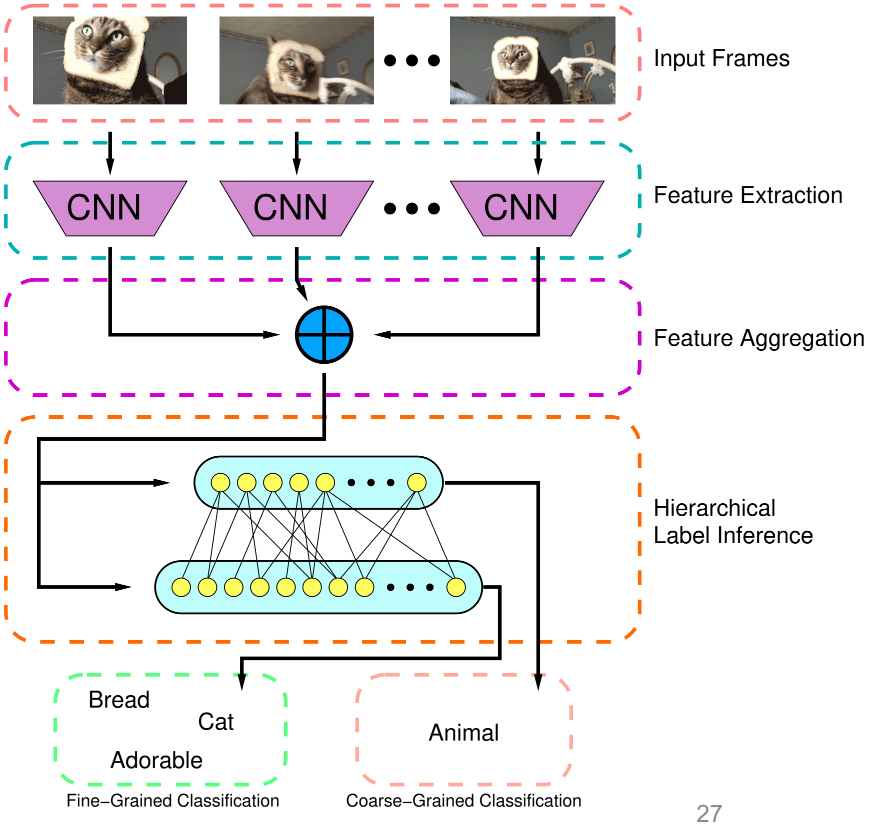
- Inception V3 frame features
- Neural network audio features



Nauata, Smith, Mori, Hierarchical Label Inference for Video Classification CVPR Workshops 2017

Results

Method	mAP / gAP	
	YouTube-8M v1	YouTube-8M v2
LSTM [Abu El Haija et al.]	26.6 / N/A	
Logistic regression [Abu El Haija et al.]	28.1 / N/A	
CNN features	27.98 / 60.34	36.84 / 70.31
BINN	31.18 / 64.74	**40.19 / 76.33**



Summary

- Inference in structured label space

- Relations within and across levels of a label space

- Model positive and negative correlations between labels in end-to-end trainable model

Desiderata for Activity Recognition Models

Label structure

Hu et al., CVPR 16
Deng et al., CVPR 16
Nauata et al., CVPRW 17
Deng et al., CVPR 17

Temporal structure

Yeung et al., CVPR 16
Yeung et al., IJCV 17
He et al., WACV 18
Chen et al., ICCVW 17

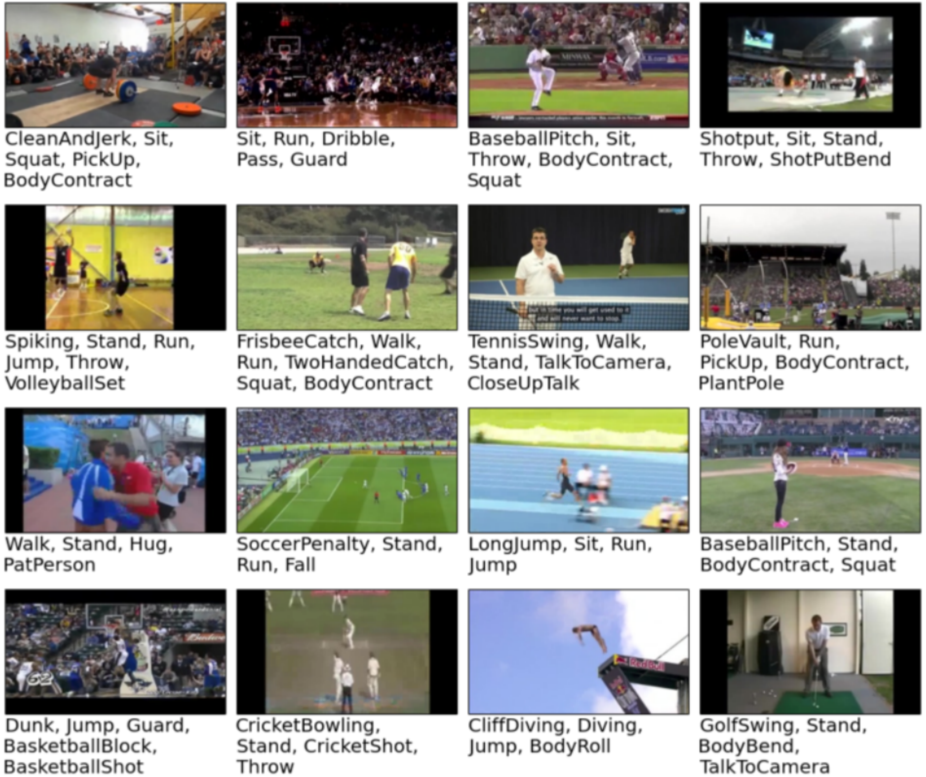
Group structure

Ibrahim et al., CVPR 16
Mehrasa et al., arXiv 17
Khodabandeh et al., arXiv 17
Lan et al. CVPR 12

MultiTHUMOS

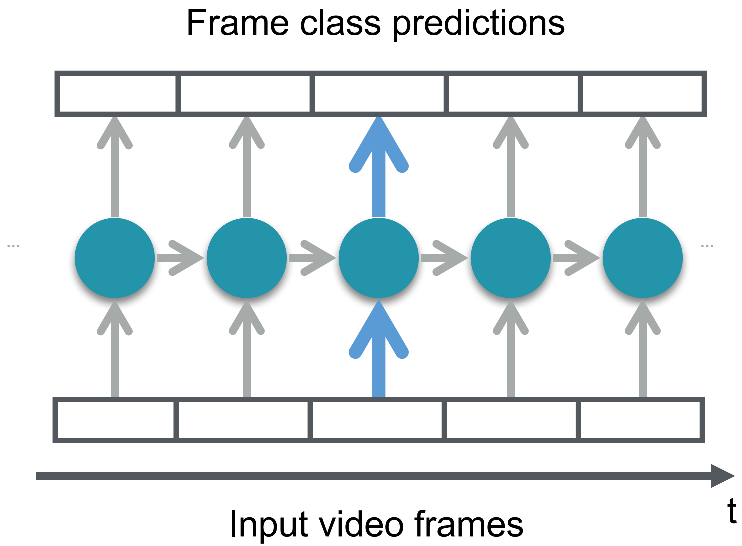
Dense labels on 30 hours of THUMOS'14

	THUMOS	MultiTHUMOS
Annotations	6,365	**38,690**
Classes	20	**65**
Labels per frame	0.3	**1.5**
Classes per video	1.1	**10.5**
Max actions per frame	2	**9**
Max actions per video	3	**25**



Yeung, Russakovsky, Jin, Andriluka, Mori, Fei-Fei. Every Moment Counts: Dense Detailed Labeling of Actions in Complex Videos. IJCV 2017.

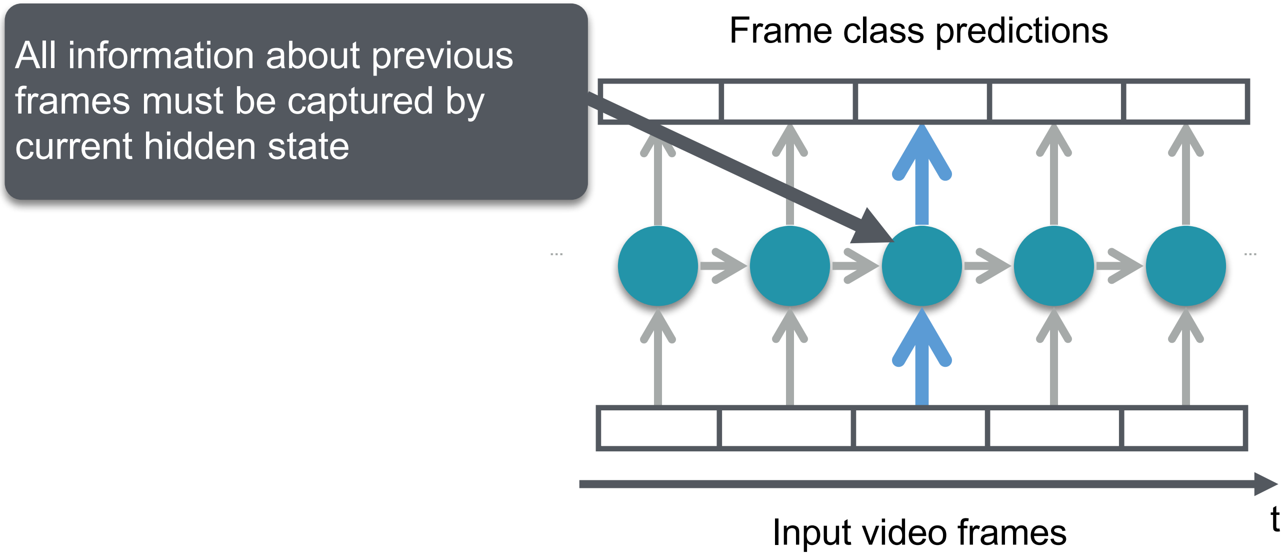
Modeling dense, multilabel actions



Standard LSTM: Single input, single output
Hochreiter 1997, Donahue 2014

Yeung, Russakovsky, Jin, Andriluka, Mori, Fei-Fei. Every Moment Counts: Dense Detailed Labeling of Actions in Complex Videos. IJCV 2017.

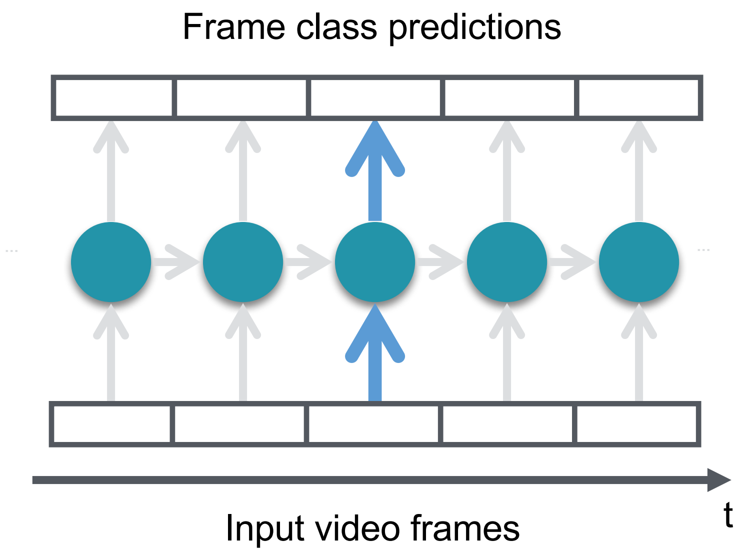
Modeling dense, multilabel actions



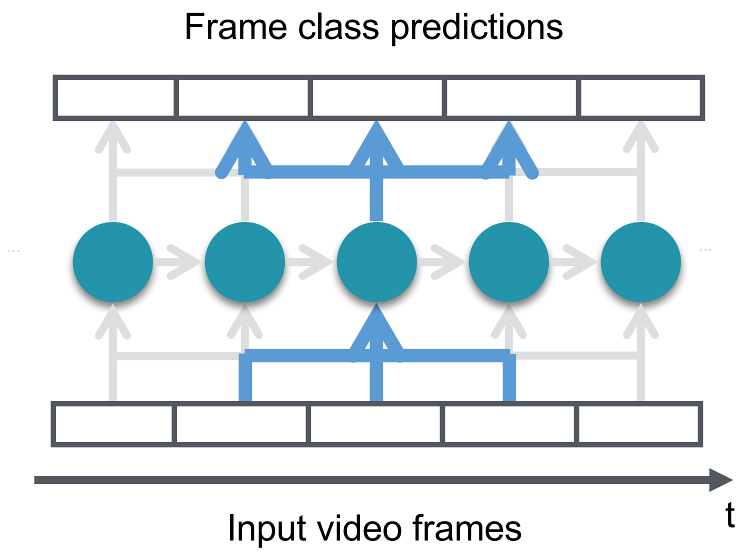
Standard LSTM: Single input, single output
Hochreiter 1997, Donahue 2014

Yeung, Russakovsky, Jin, Andriluka, Mori, Fei-Fei. Every Moment Counts: Dense Detailed Labeling of Actions in Complex Videos. IJCV 2017.

MultiLSTM



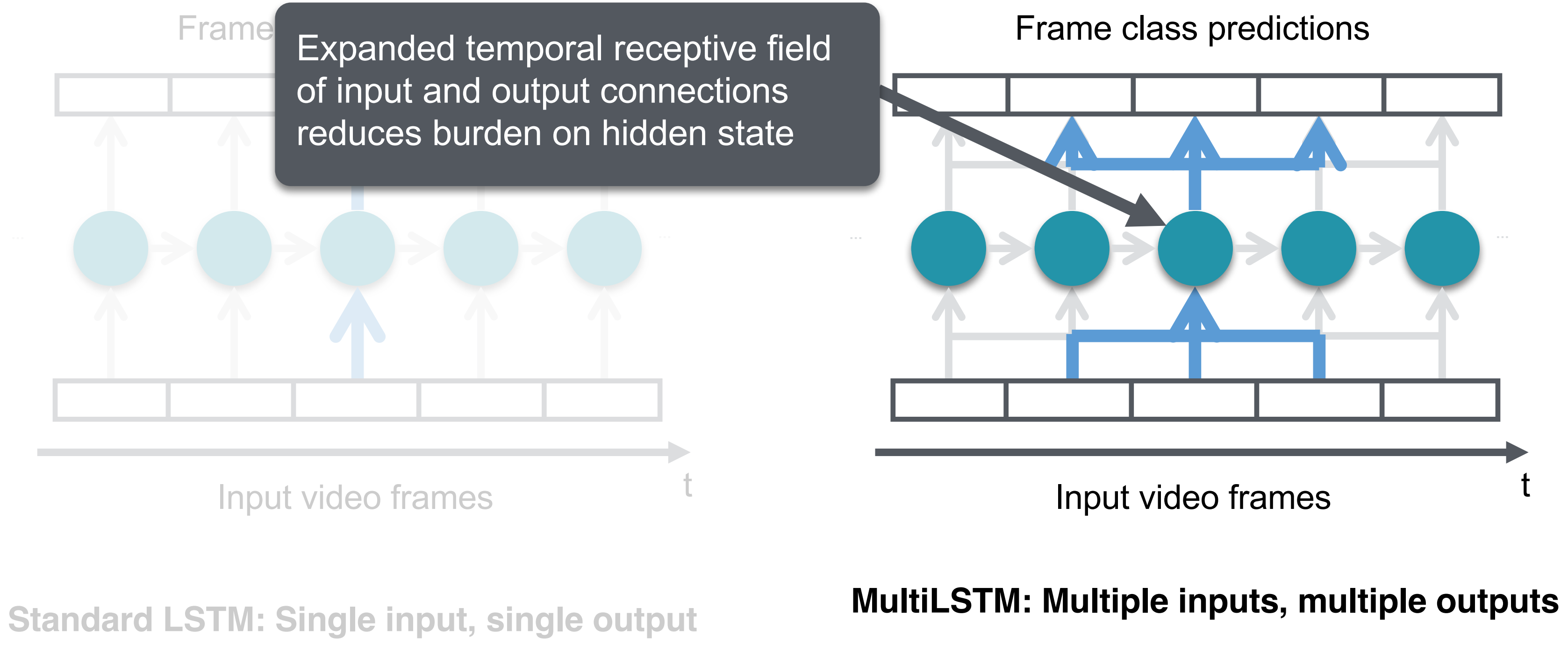
Standard LSTM: Single input, single output



MultiLSTM: Multiple inputs, multiple outputs

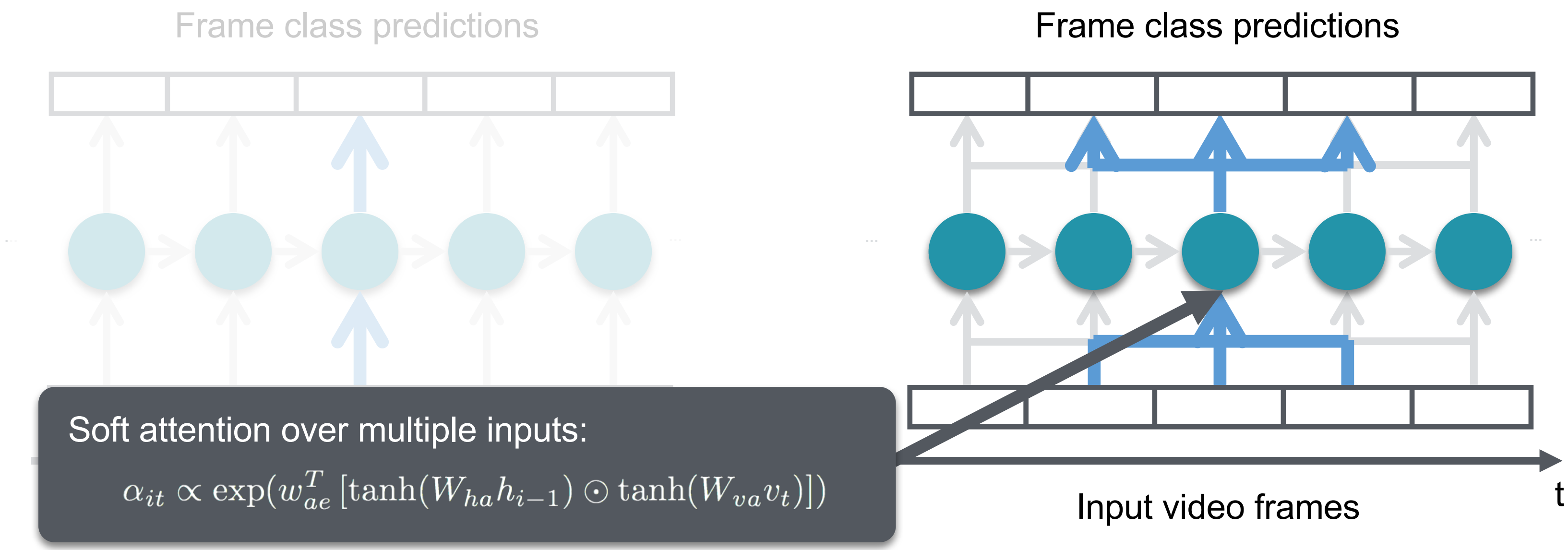
Yeung, Russakovsky, Jin, Andriluka, Mori, Fei-Fei. Every Moment Counts: Dense Detailed Labeling of Actions in Complex Videos. IJCV 2017.

MultiLSTM



Yeung, Russakovsky, Jin, Andriluka, Mori, Fei-Fei. Every Moment Counts: Dense Detailed Labeling of Actions in Complex Videos. IJCV 2017.

MultiLSTM



Soft attention over multiple inputs:

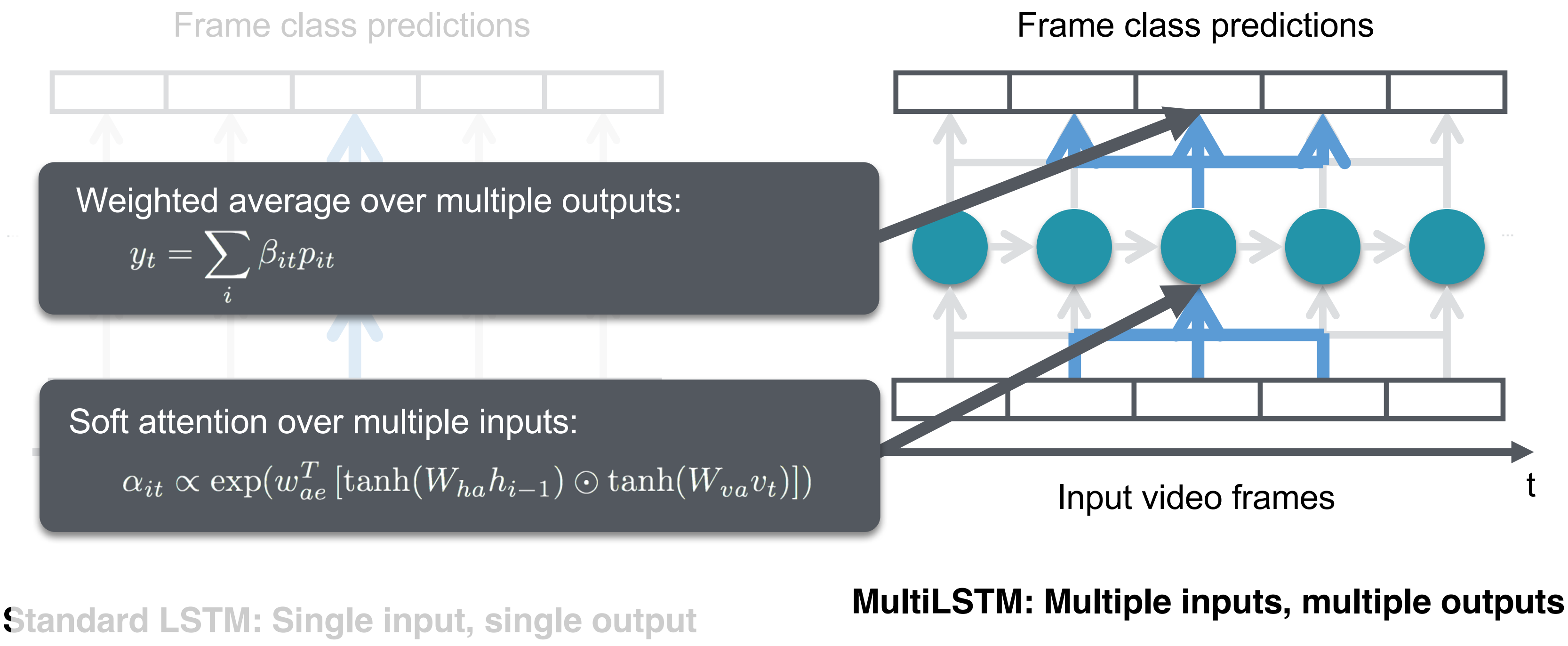
$$\alpha_{it} \propto \exp(w_{ae}^T [\tanh(W_{ha} h_{i-1}) \odot \tanh(W_{va} v_t)])$$

Standard LSTM: Single input, single output

MultiLSTM: Multiple inputs, multiple outputs

Yeung, Russakovsky, Jin, Andriluka, Mori, Fei-Fei. Every Moment Counts: Dense Detailed Labeling of Actions in Complex Videos. IJCV 2017.

MultiLSTM



Yeung, Russakovsky, Jin, Andriluka, Mori, Fei-Fei. Every Moment Counts: Dense Detailed Labeling of Actions in Complex Videos. IJCV 2017.

MultiLSTM

Multilabel loss (per-class binary cross entropy):

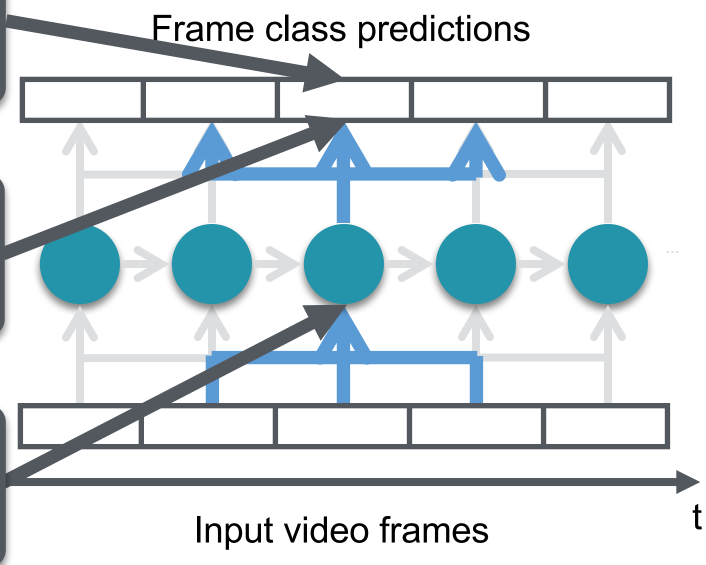
$$L(\mathbf{y}|\mathbf{x}) = \sum_{t,c} z_{tc} \log(\sigma(y_{tc})) + (1 - z_{tc}) \log(1 - \sigma(y_{tc}))$$

Weighted average over multiple outputs:

$$y_t = \sum_i \beta_{it} p_{it}$$

Soft attention over multiple inputs:

$$\alpha_{it} \propto \exp(w_{ae}^T [\tanh(W_{ha} h_{i-1}) \odot \tanh(W_{va} v_t)])$$



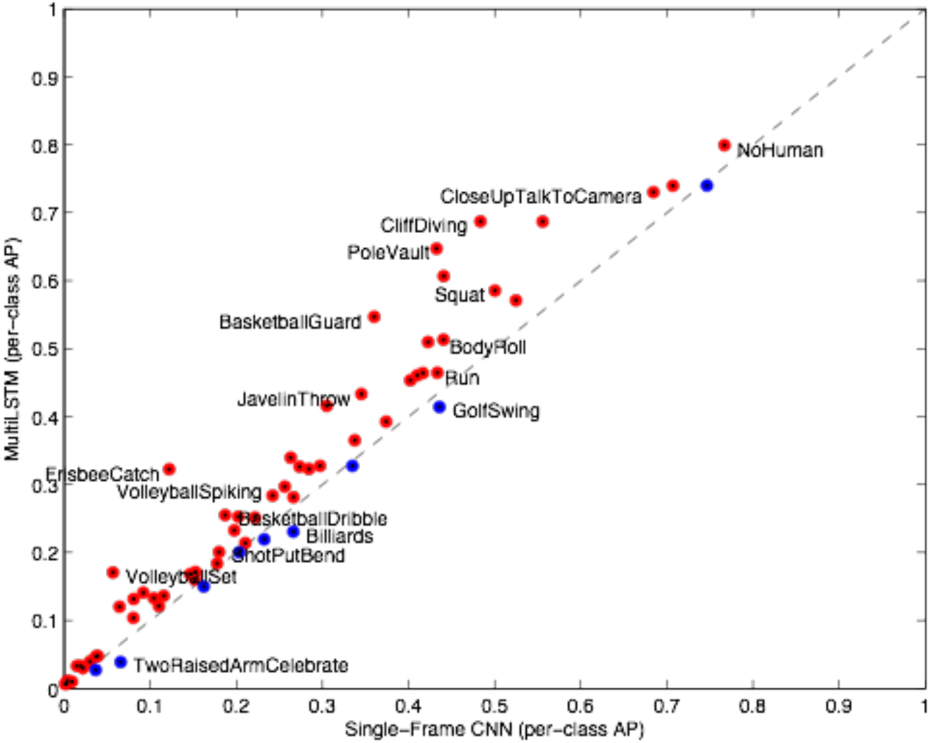
Standard LSTM: Single input, single output

MultiLSTM: Multiple inputs, multiple outputs

Yeung, Russakovsky, Jin, Andriluka, Mori, Fei-Fei. Every Moment Counts: Dense Detailed Labeling of Actions in Complex Videos. IJCV 2017.

MultiLSTM

Model	THUMOS mAP	MultiTHUMOS mAP
IDT	13.6	13.3
Single-frame CNN	34.7	25.4
Two-stream CNN	36.2	27.6
LSTM	39.3	28.1
LSTM+i	39.5	28.7
LSTM+i+a	39.7	29.1
MultiLSTM	**41.3**	**29.7**



Yeung, Russakovsky, Jin, Andriluka, Mori, Fei-Fei. Every Moment Counts: Dense Detailed Labeling of Actions in Complex Videos. IJCV 2017.

Retrieving sequential and co-occurring actions

Sequential actions

Pass, then Shot

Throw, then One-handed catch

Jump, then Fall

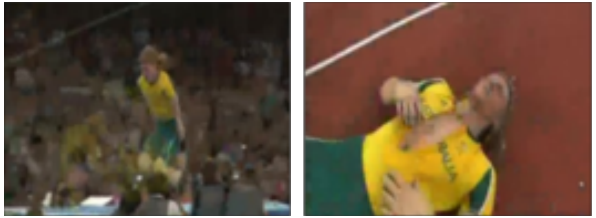
Yeung, Russakovsky, Jin, Andriluka, Mori, Fei-Fei. Every Moment Counts: Dense Detailed Labeling of Actions in Complex Videos. IJCV 2017.

Retrieving sequential and co-occurring actions

Sequential actions

Pass, then Shot

Throw, then One-handed catch



Jump, then Fall

Co-occurring actions

Dive & No Bodyroll

Dive & Bodyroll

Shot & Guard

Shot & No Guard

Talk & Sit

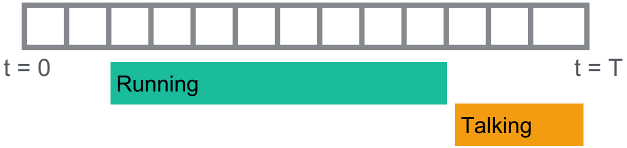
Talk & Stand

Yeung, Russakovsky, Jin, Andriluka, Mori, Fei-Fei. Every Moment Counts: Dense Detailed Labeling of Actions in Complex Videos. IJCV 2017.

Task: action detection

Input

Output

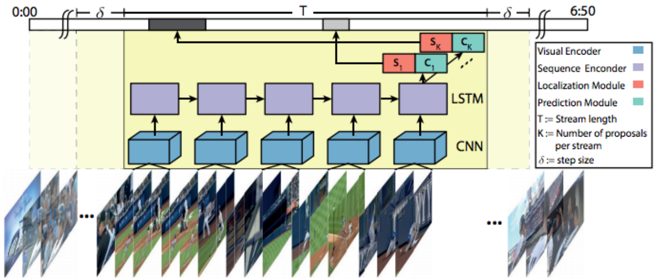


Yeung, Russakovsky, Mori, Fei-Fei. End-to-end Learning of Action Detection from Frame Glimpses in Videos. CVPR 2016.

Dominant paradigm: Dense processing

Standard in THUMOS challenge action detection entries
Oneata et al. 2014
Wang et al. 2014
Oneata et al. 2014
Yuan et al. 2015

Sliding windows

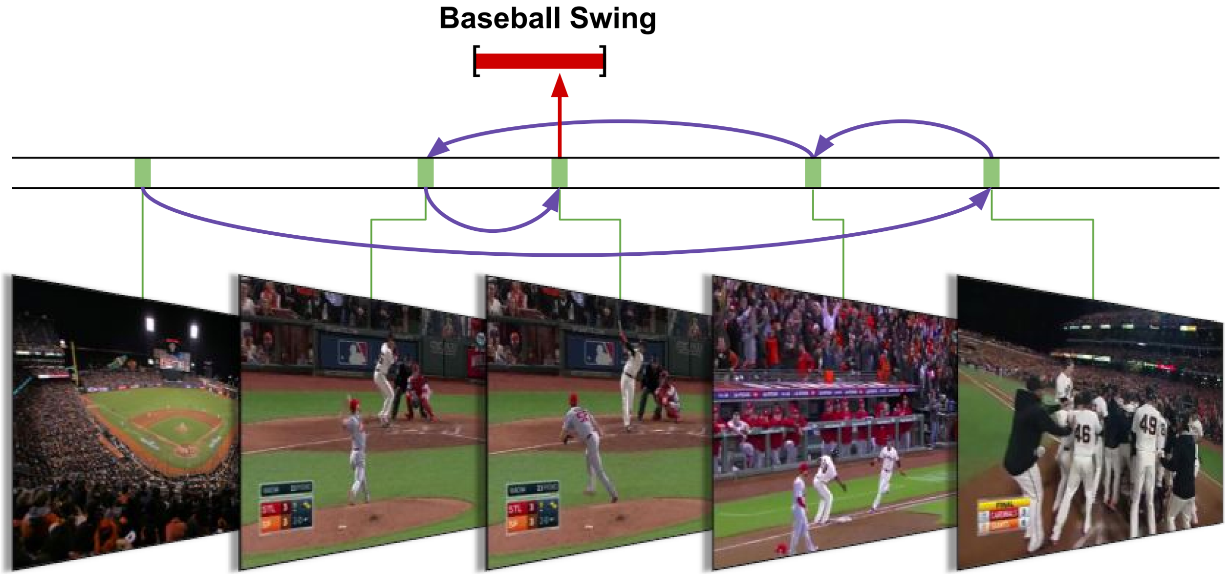


Gkioxari and Malik 2015
Yu et al. 2015
Escorcia et al. 2016
Peng and Schmid 2016
He et al. 2018

Action proposals

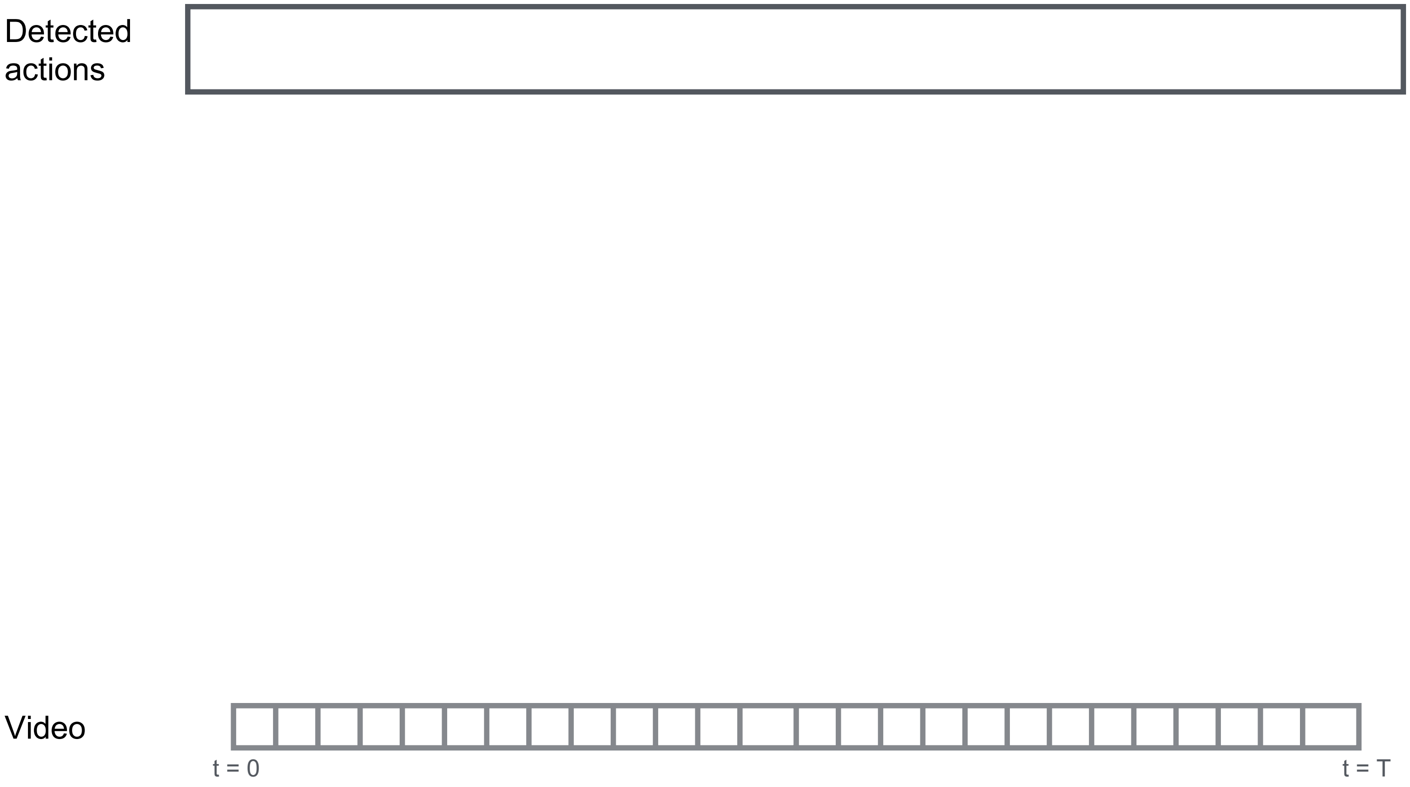
Yeung, Russakovsky, Mori, Fei-Fei. End-to-end Learning of Action Detection from Frame Glimpses in Videos. CVPR 2016.

Efficiently detecting actions



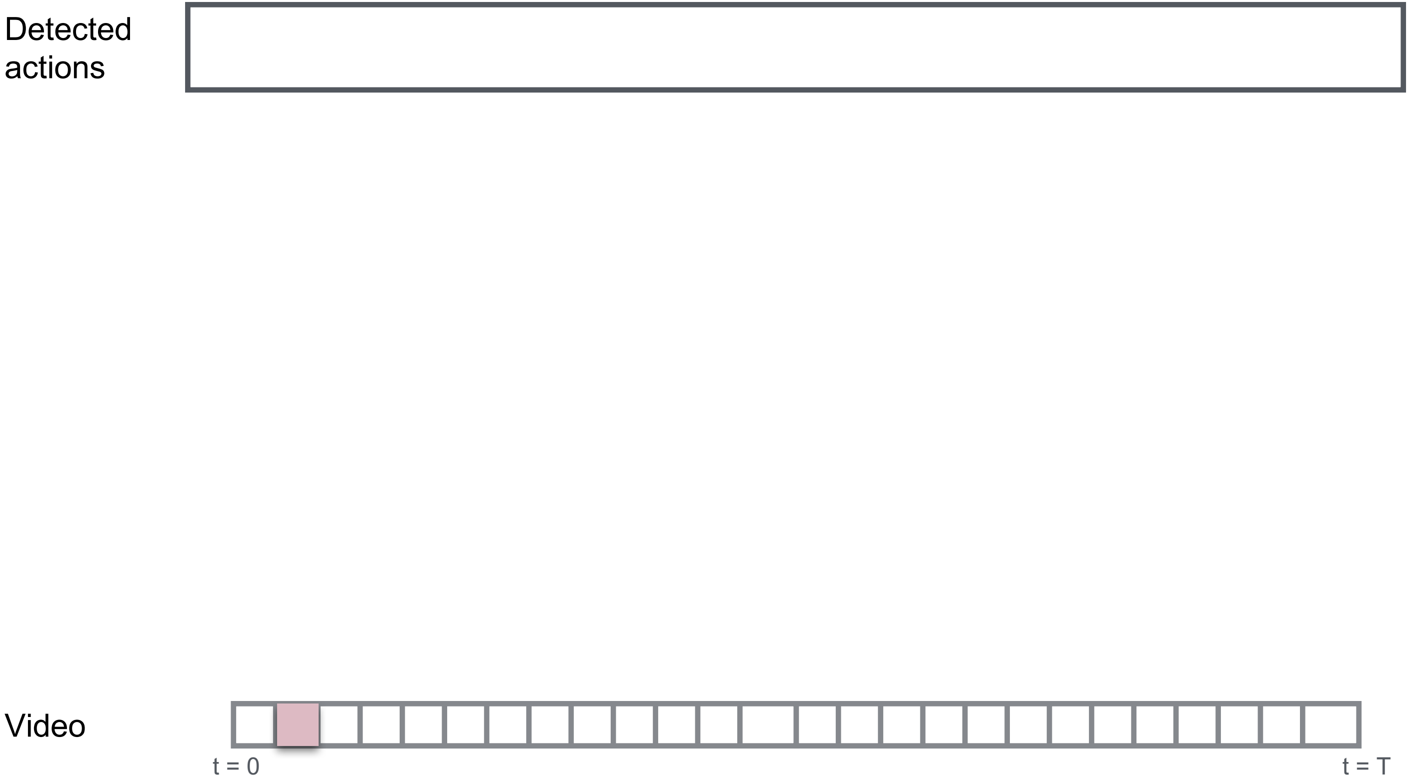
Yeung, Russakovsky, Mori, Fei-Fei. End-to-end Learning of Action Detection from Frame Glimpses in Videos. CVPR 2016.

Our model for efficient action detection



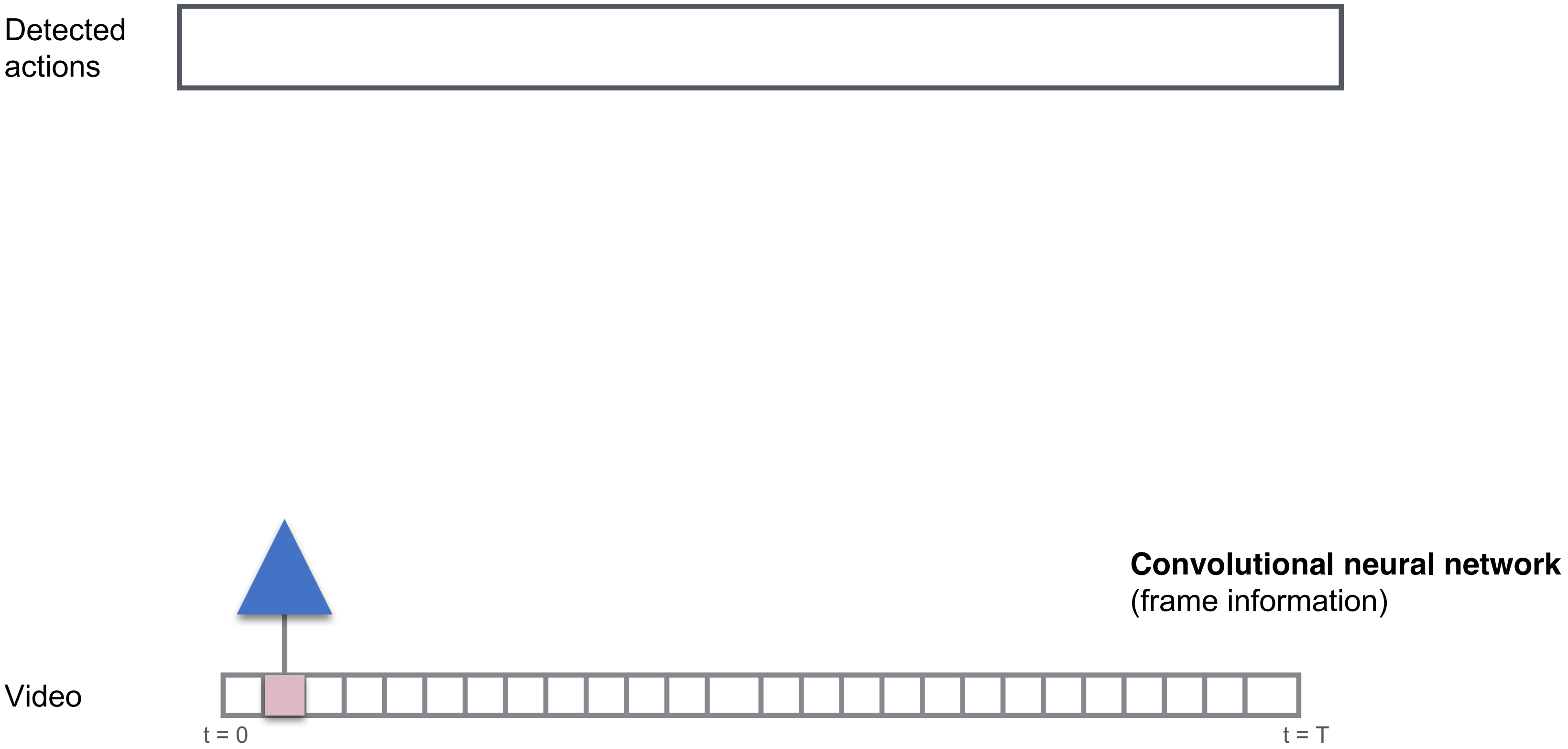
Yeung, Russakovsky, Mori, Fei-Fei. End-to-end Learning of Action Detection from Frame Glimpses in Videos. CVPR 2016.

Our model for efficient action detection



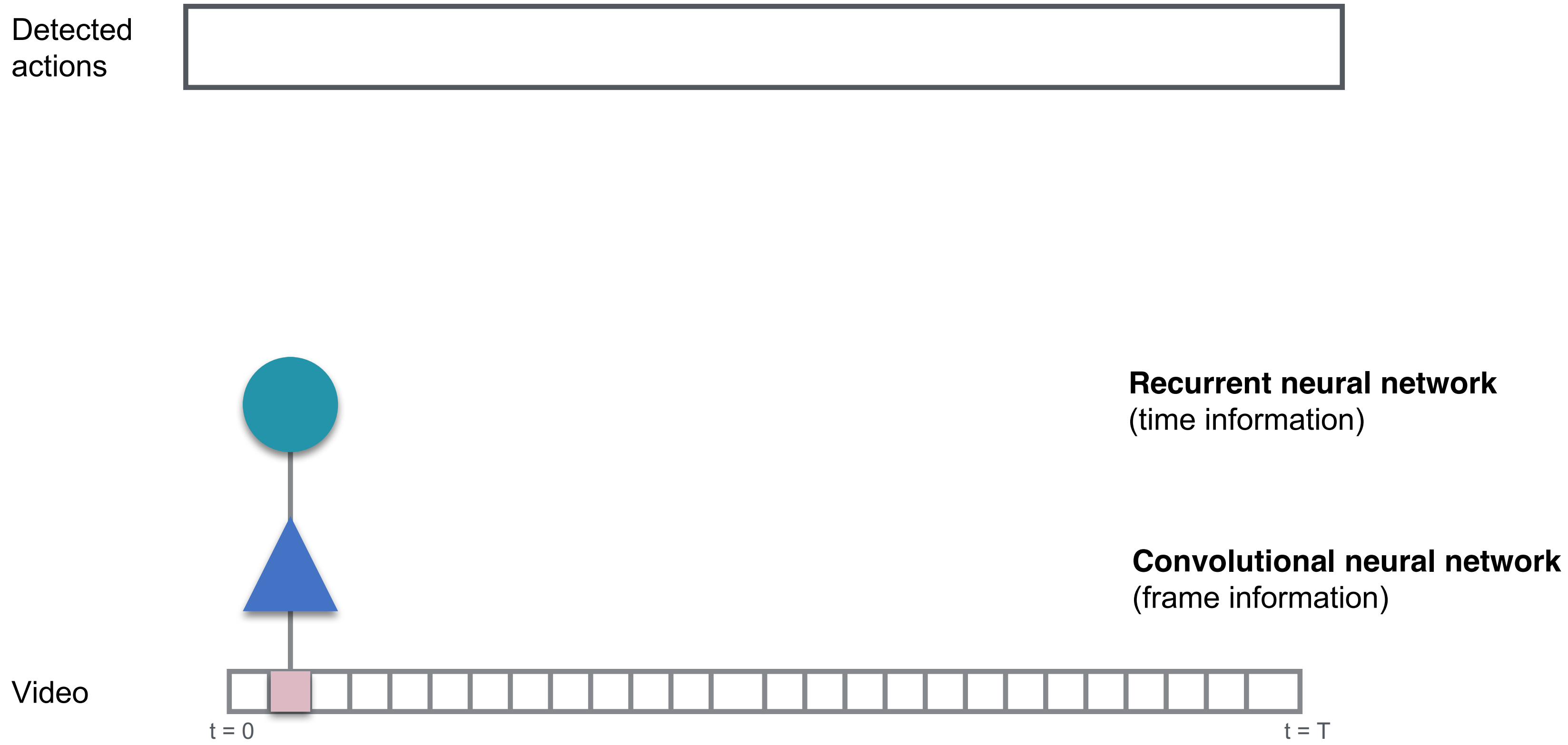
Yeung, Russakovsky, Mori, Fei-Fei. End-to-end Learning of Action Detection from Frame Glimpses in Videos. CVPR 2016.

Our model for efficient action detection



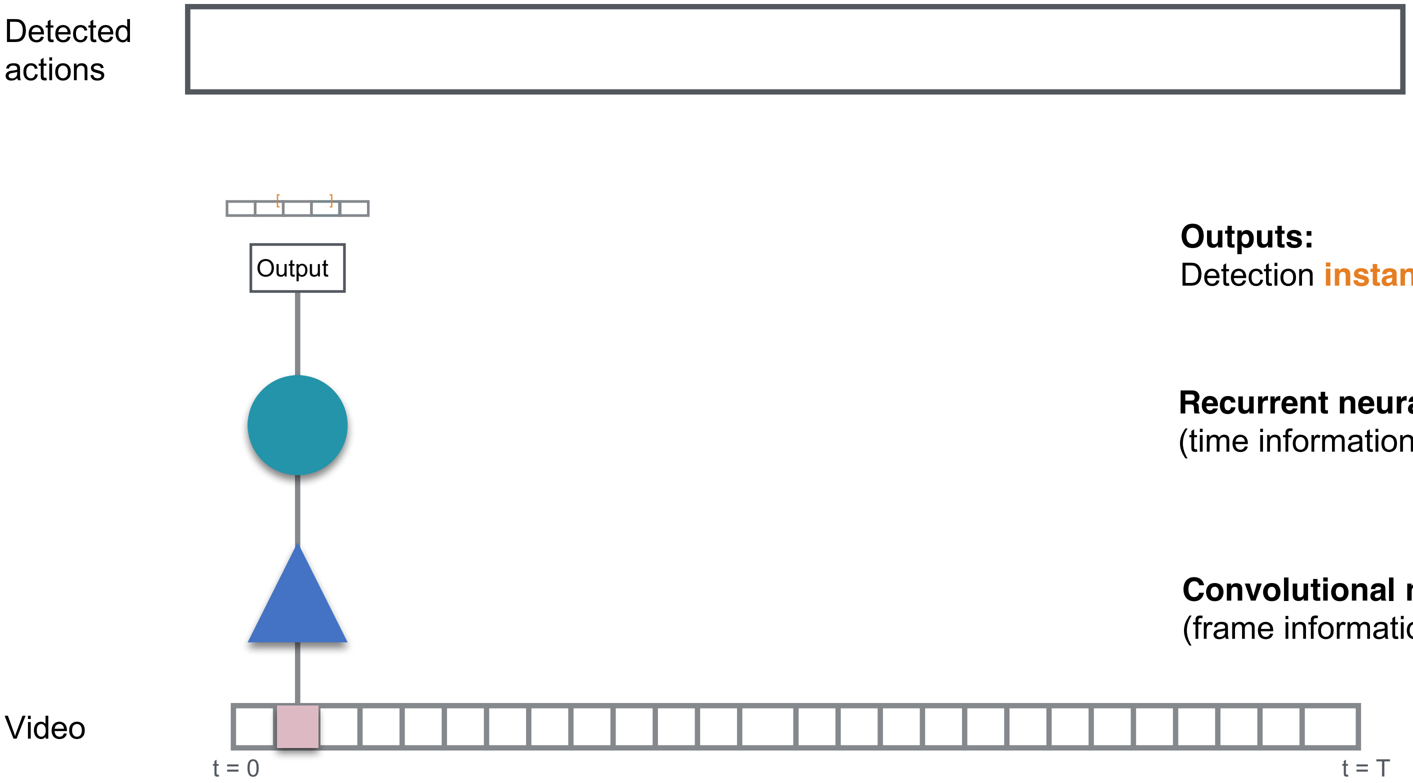
Yeung, Russakovsky, Mori, Fei-Fei. End-to-end Learning of Action Detection from Frame Glimpses in Videos. CVPR 2016.

Our model for efficient action detection



Yeung, Russakovsky, Mori, Fei-Fei. End-to-end Learning of Action Detection from Frame Glimpses in Videos. CVPR 2016.

Our model for efficient action detection



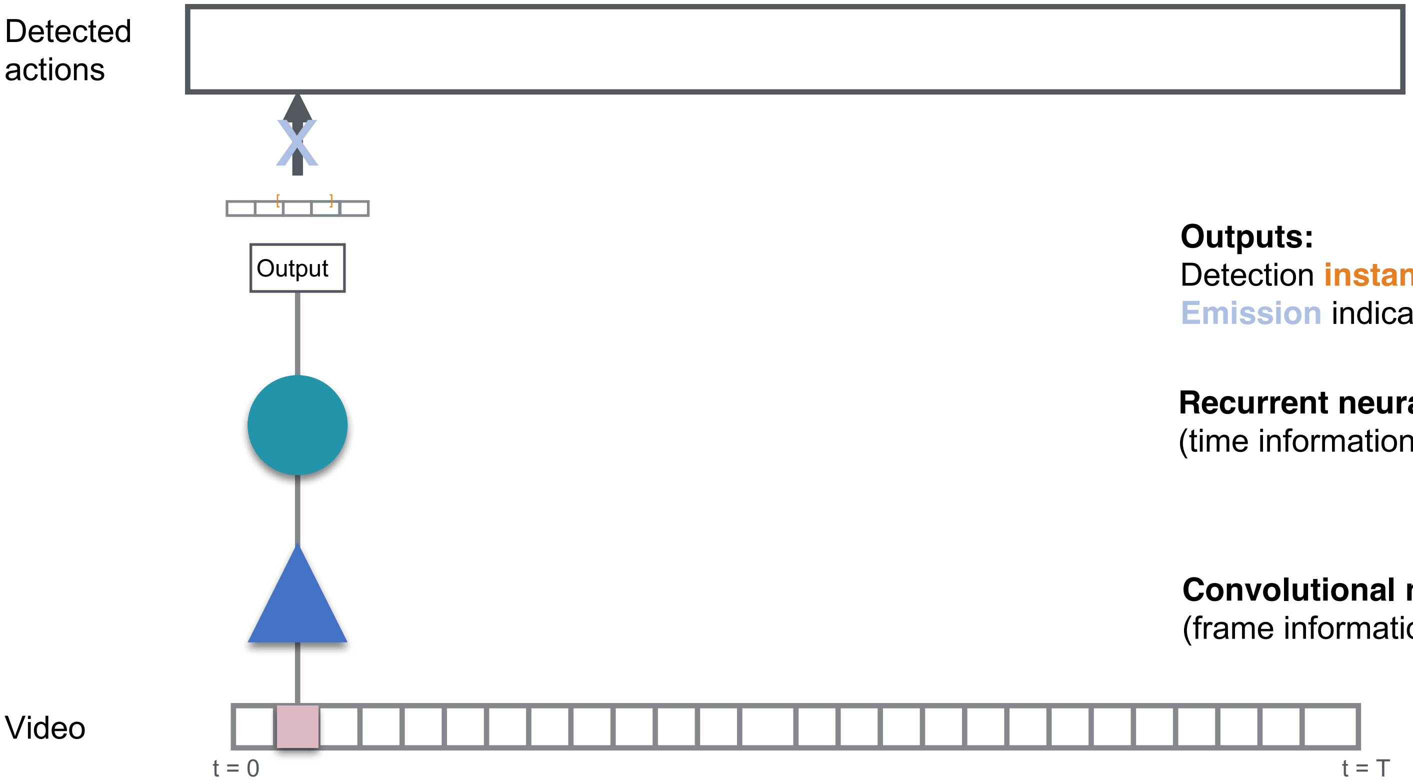
Outputs:
Detection **instance hypothesis** [start, end]

Recurrent neural network
(time information)

Convolutional neural network
(frame information)

Yeung, Russakovsky, Mori, Fei-Fei. End-to-end Learning of Action Detection from Frame Glimpses in Videos. CVPR 2016.

Our model for efficient action detection



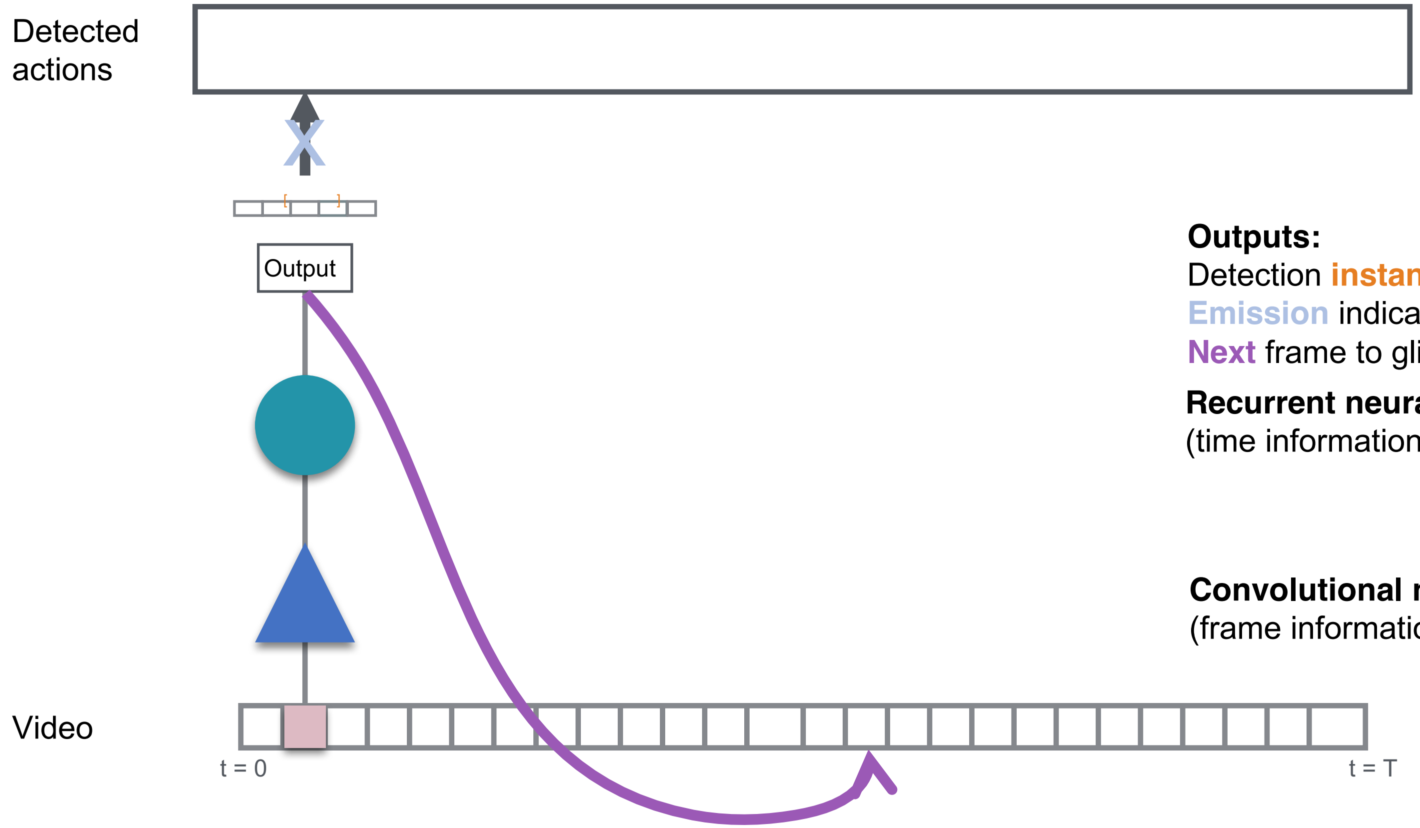
Outputs:
Detection instance hypothesis [start, end]
Emission indicator

Recurrent neural network
(time information)

Convolutional neural network
(frame information)

Yeung, Russakovsky, Mori, Fei-Fei. End-to-end Learning of Action Detection from Frame Glimpses in Videos. CVPR 2016.

Our model for efficient action detection



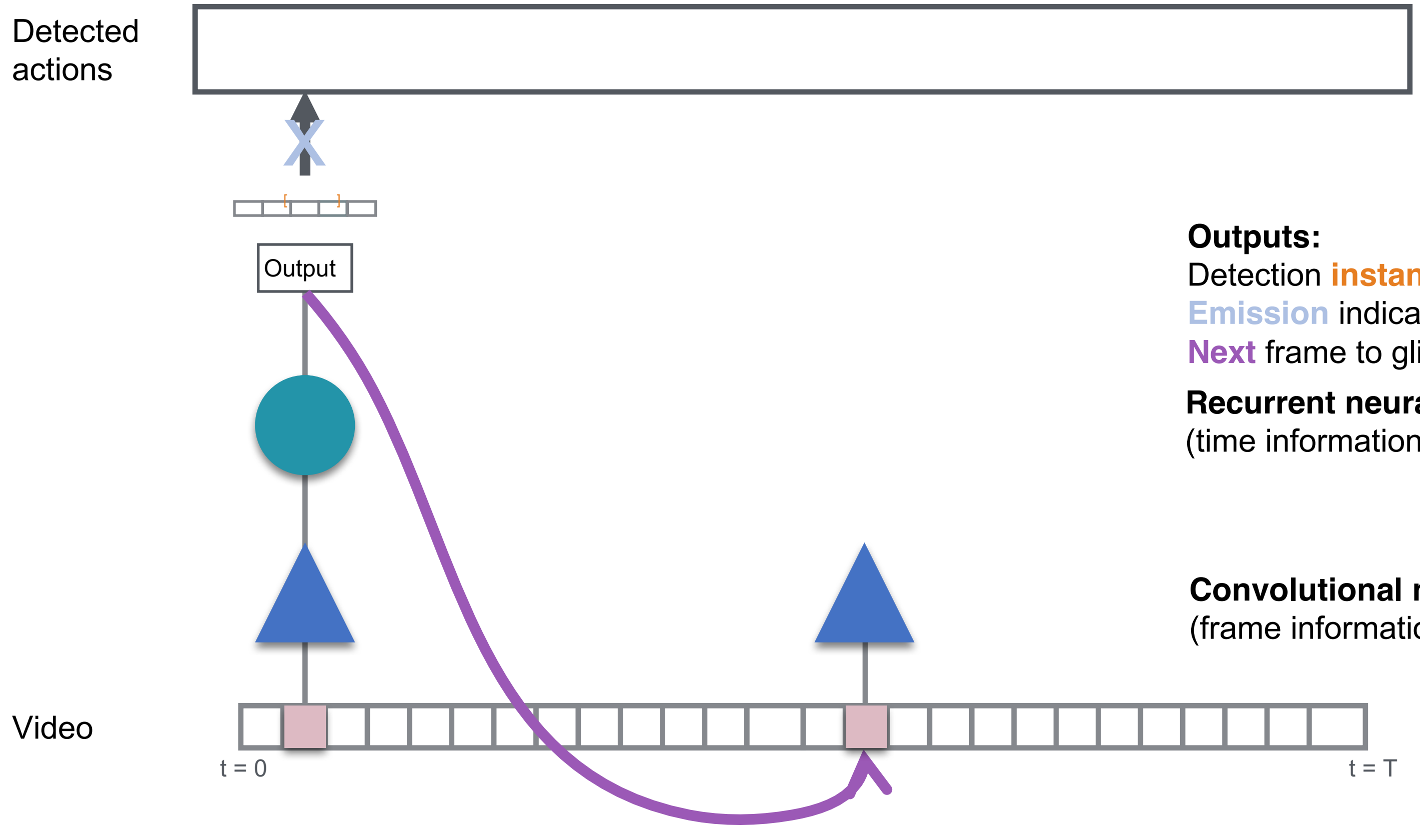
Outputs:
Detection instance hypothesis [start, end]
Emission indicator
Next frame to glimpse

Recurrent neural network
(time information)

Convolutional neural network
(frame information)

Yeung, Russakovsky, Mori, Fei-Fei. End-to-end Learning of Action Detection from Frame Glimpses in Videos. CVPR 2016.

Our model for efficient action detection



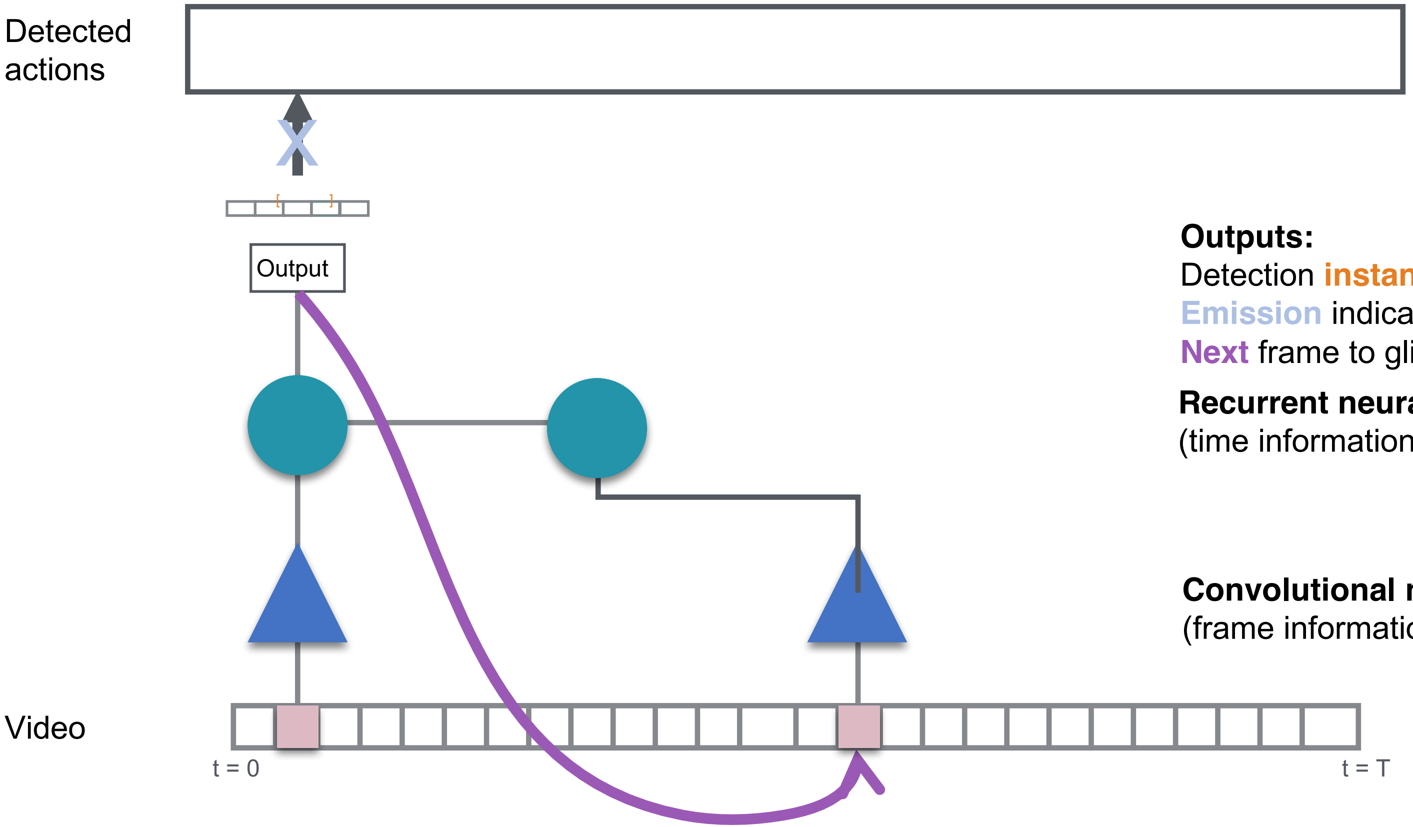
Outputs:
Detection **instance hypothesis** [start, end]
Emission indicator
Next frame to glimpse

Recurrent neural network
(time information)

Convolutional neural network
(frame information)

Yeung, Russakovsky, Mori, Fei-Fei. End-to-end Learning of Action Detection from Frame Glimpses in Videos. CVPR 2016.

Our model for efficient action detection



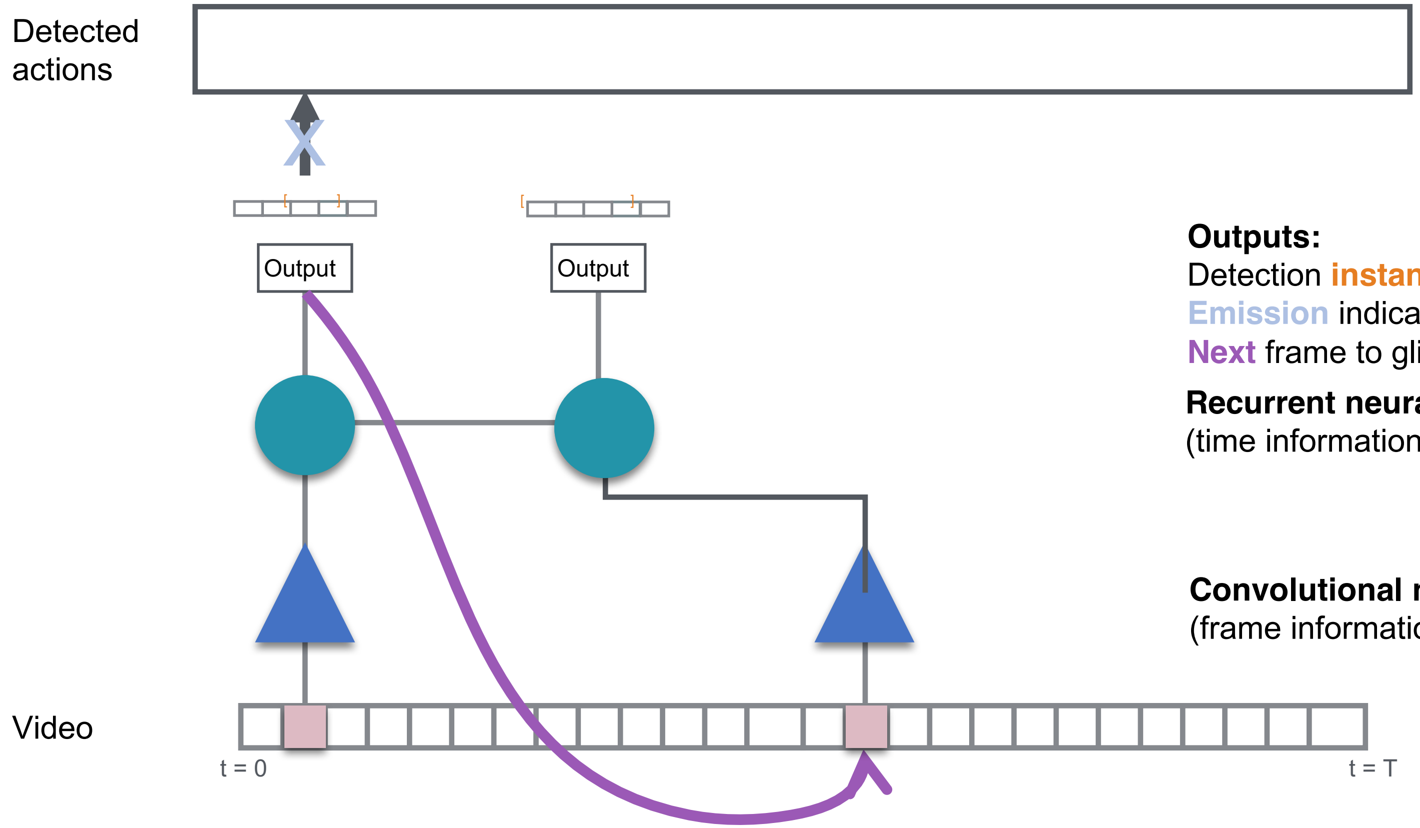
Outputs:
Detection instance hypothesis [start, end]
Emission indicator
Next frame to glimpse

Recurrent neural network
(time information)

Convolutional neural network
(frame information)

Yeung, Russakovsky, Mori, Fei-Fei. End-to-end Learning of Action Detection from Frame Glimpses in Videos. CVPR 2016.

Our model for efficient action detection



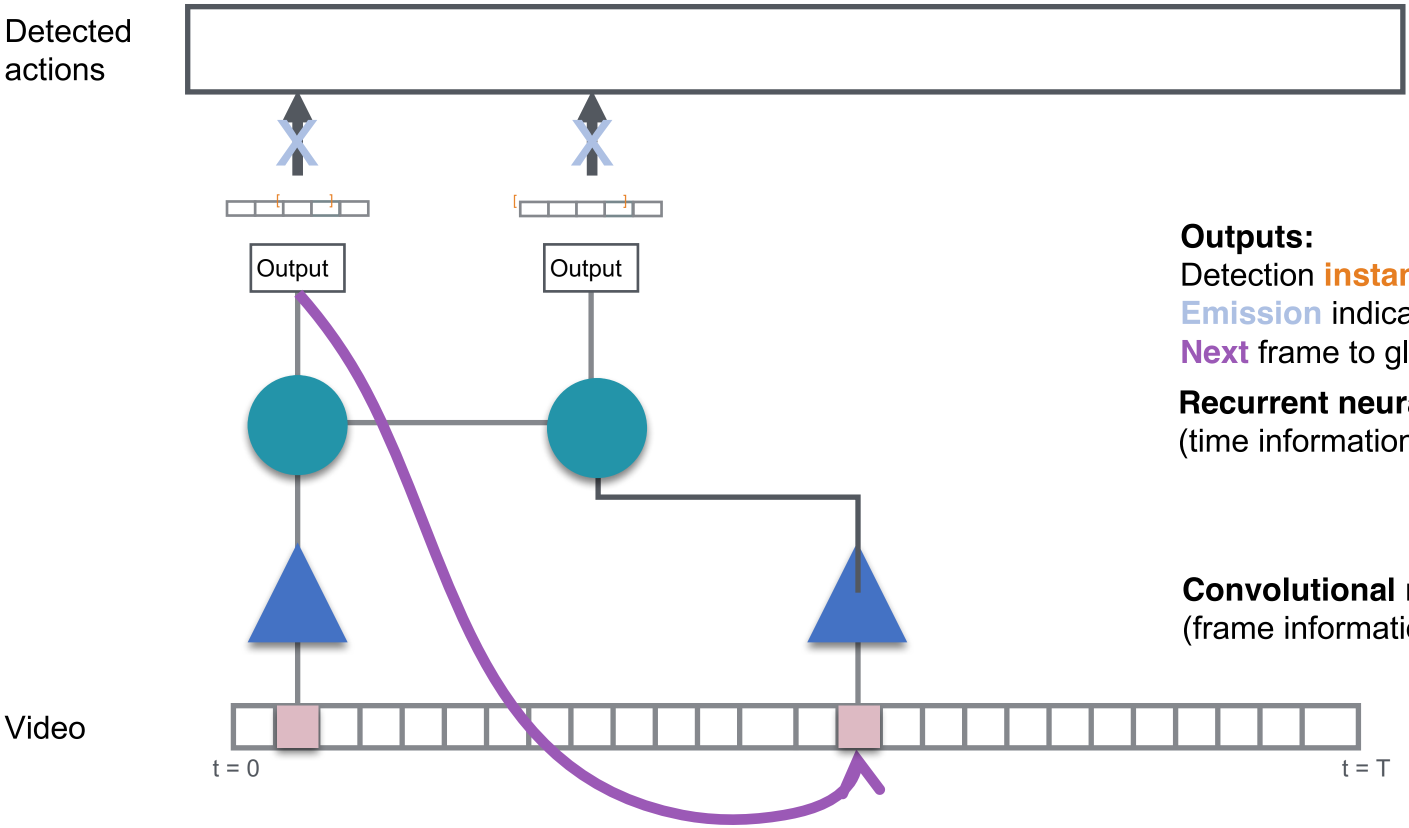
Outputs:
Detection **instance hypothesis** [start, end]
Emission indicator
Next frame to glimpse

Recurrent neural network
(time information)

Convolutional neural network
(frame information)

Yeung, Russakovsky, Mori, Fei-Fei. End-to-end Learning of Action Detection from Frame Glimpses in Videos. CVPR 2016.

Our model for efficient action detection



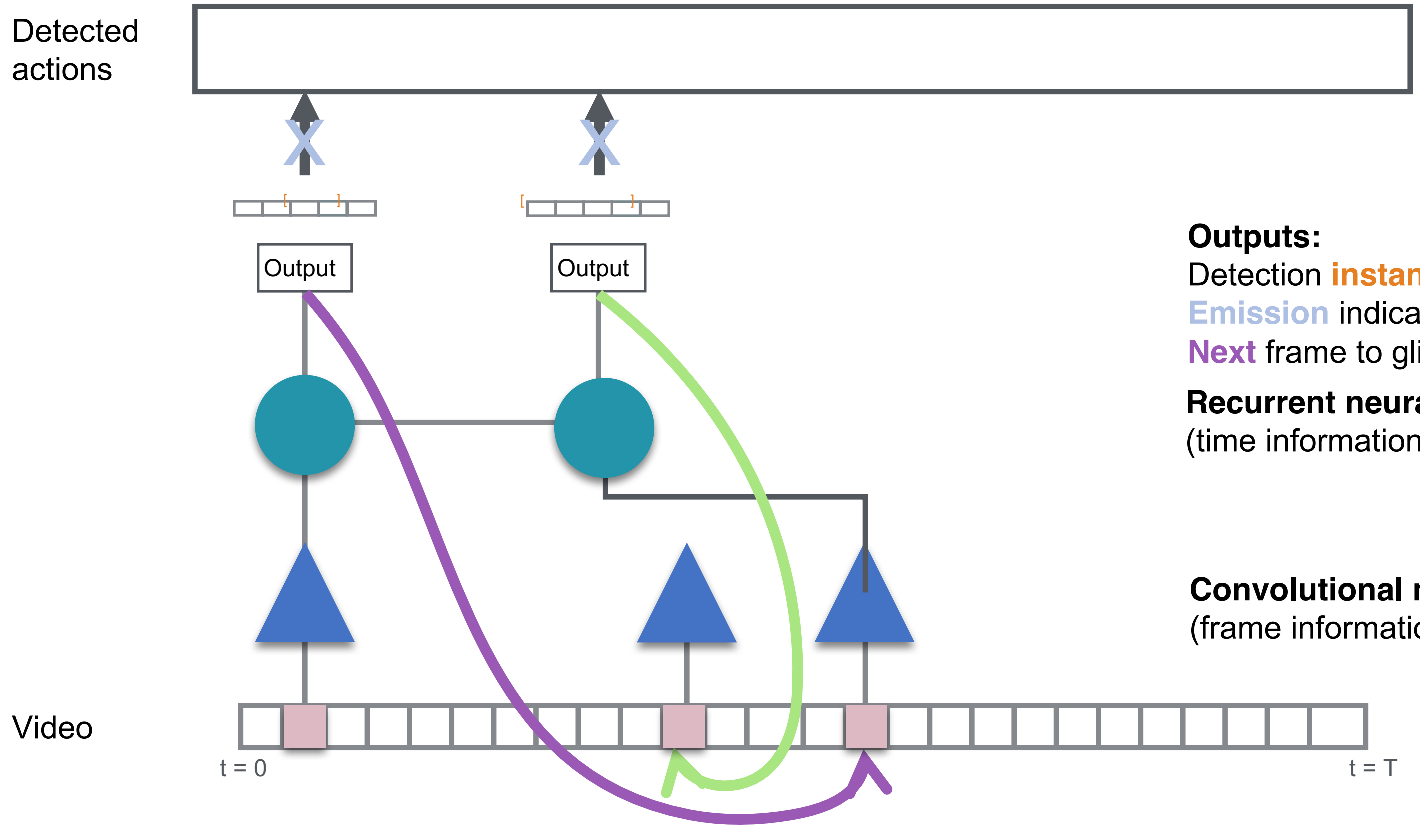
Outputs:
Detection instance hypothesis [start, end]
Emission indicator
Next frame to glimpse

Recurrent neural network
(time information)

Convolutional neural network
(frame information)

Yeung, Russakovsky, Mori, Fei-Fei. End-to-end Learning of Action Detection from Frame Glimpses in Videos. CVPR 2016.

Our model for efficient action detection



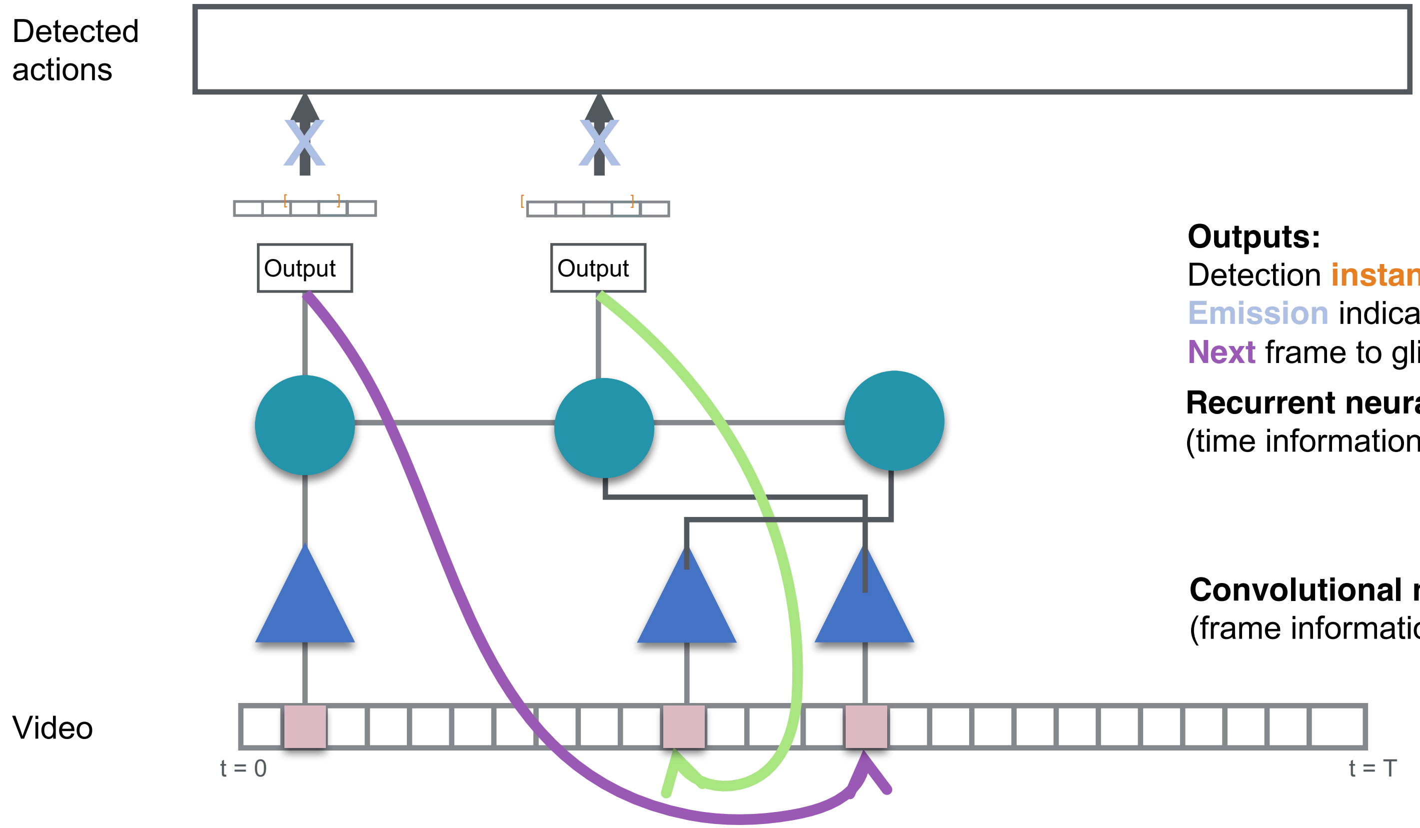
Outputs:
Detection **instance hypothesis** [start, end]
Emission indicator
Next frame to glimpse

Recurrent neural network
(time information)

Convolutional neural network
(frame information)

Yeung, Russakovsky, Mori, Fei-Fei. End-to-end Learning of Action Detection from Frame Glimpses in Videos. CVPR 2016.

Our model for efficient action detection



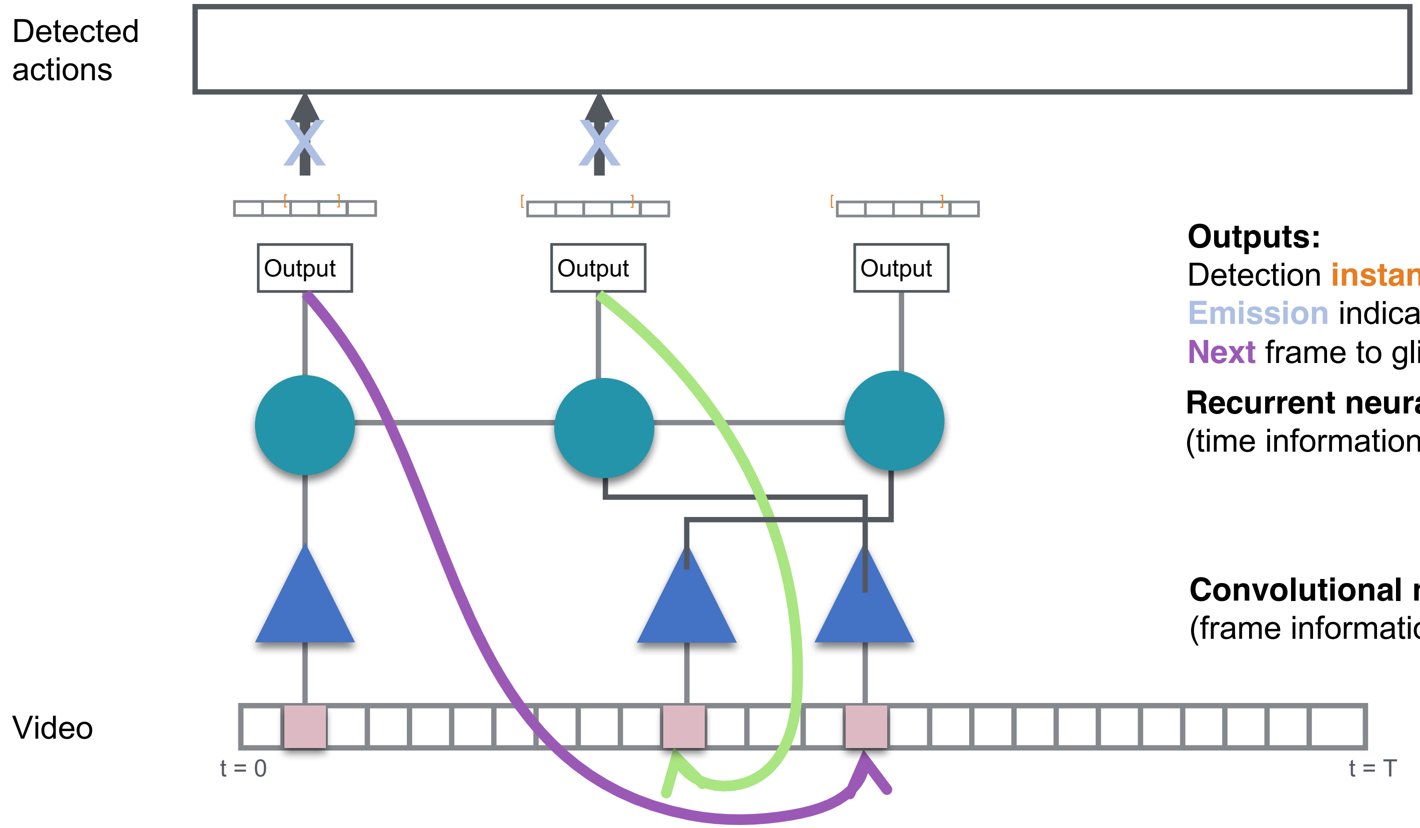
Outputs:
Detection instance hypothesis [start, end]
Emission indicator
Next frame to glimpse

Recurrent neural network
(time information)

Convolutional neural network
(frame information)

Yeung, Russakovsky, Mori, Fei-Fei. End-to-end Learning of Action Detection from Frame Glimpses in Videos. CVPR 2016.

Our model for efficient action detection



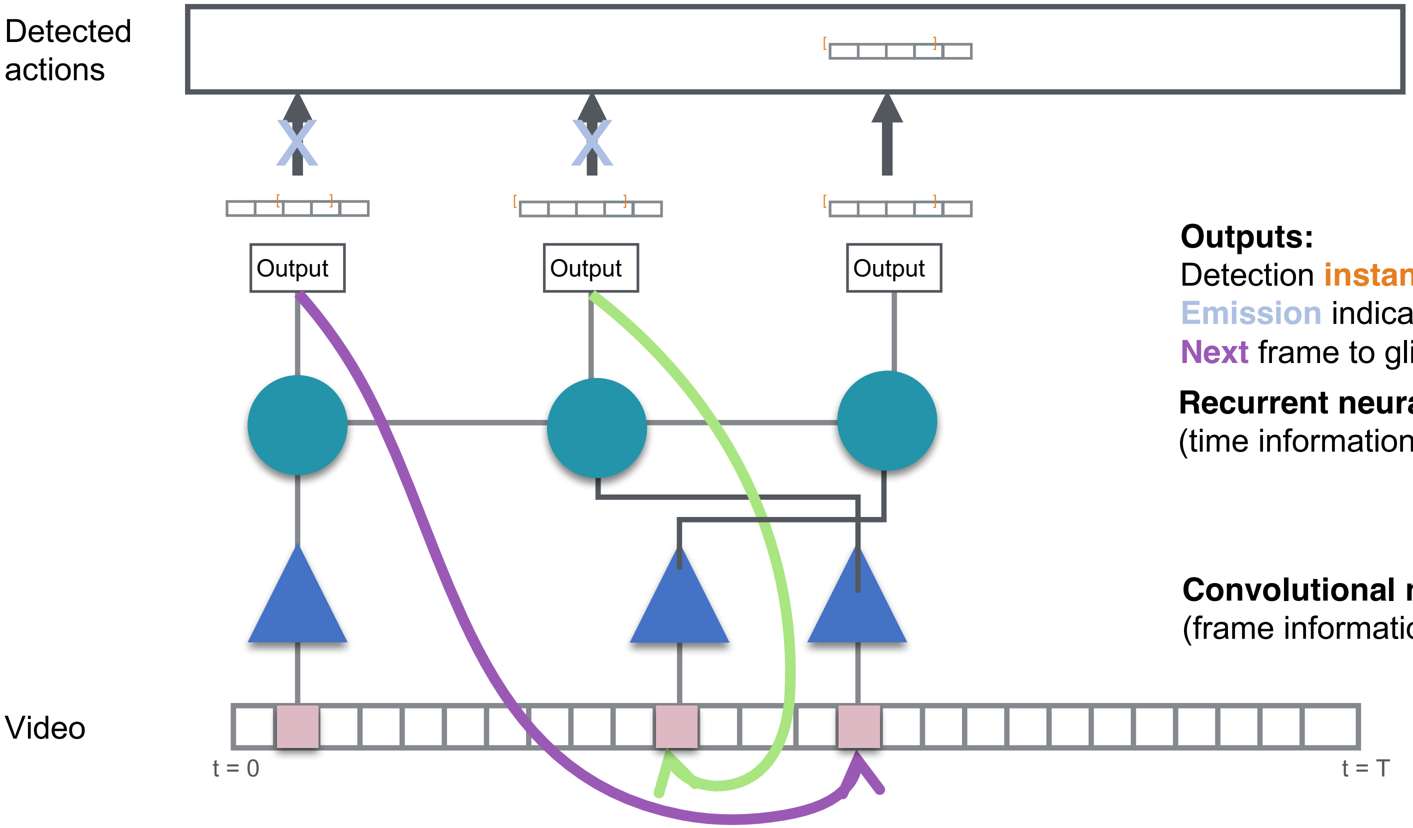
Outputs:
Detection instance hypothesis [start, end]
Emission indicator
Next frame to glimpse

Recurrent neural network
(time information)

Convolutional neural network
(frame information)

Yeung, Russakovsky, Mori, Fei-Fei. End-to-end Learning of Action Detection from Frame Glimpses in Videos. CVPR 2016.

Our model for efficient action detection



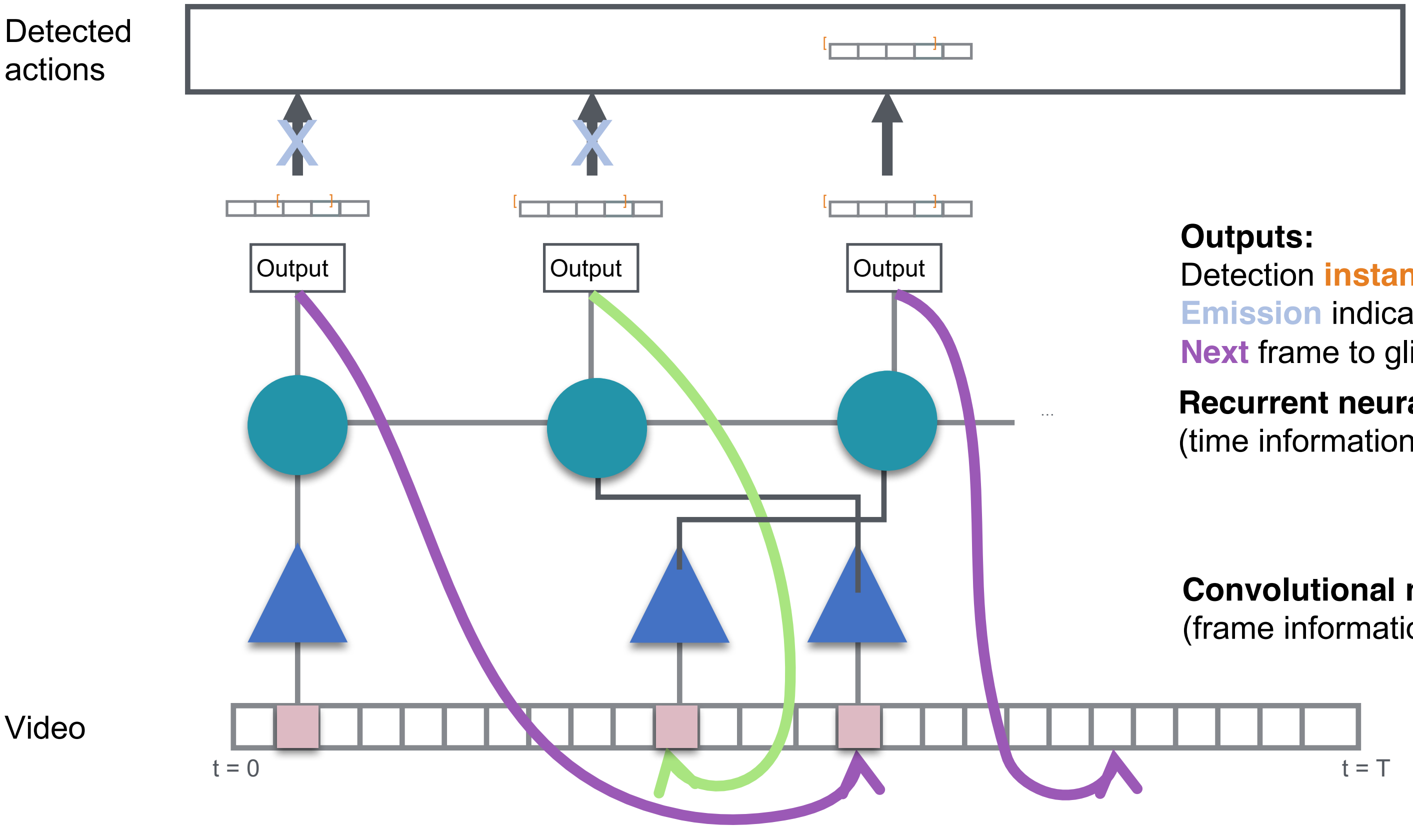
Outputs:
Detection **instance hypothesis** [start, end]
Emission indicator
Next frame to glimpse

Recurrent neural network
(time information)

Convolutional neural network
(frame information)

Yeung, Russakovsky, Mori, Fei-Fei. End-to-end Learning of Action Detection from Frame Glimpses in Videos. CVPR 2016.

Our model for efficient action detection



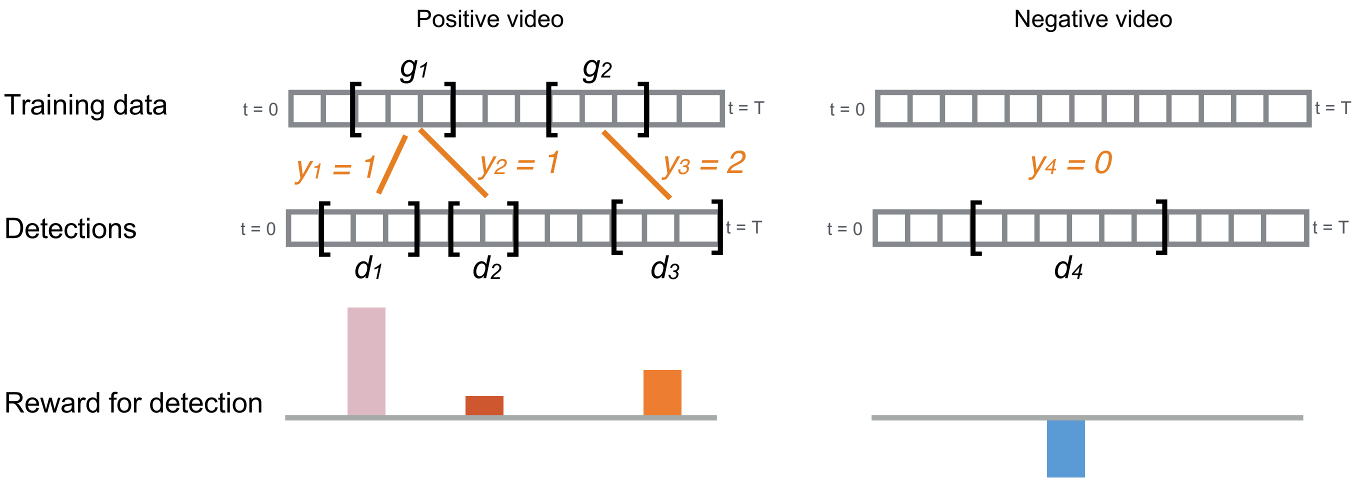
Outputs:
Detection **instance hypothesis** [start, end]
Emission indicator
Next frame to glimpse

Recurrent neural network
(time information)

Convolutional neural network
(frame information)

Yeung, Russakovsky, Mori, Fei-Fei. End-to-end Learning of Action Detection from Frame Glimpses in Videos. CVPR 2016.

Training the detection instance output



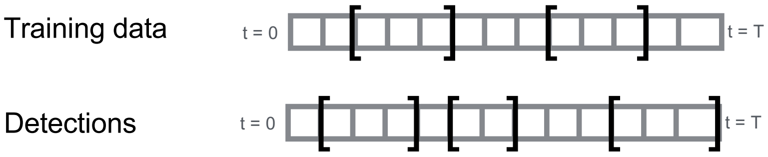
$$\mathcal{L}(D,G) = \sum_i \mathcal{L}_{cls}(d_i, y_i > 0) + \gamma \sum_{i:y_i>0} \mathcal{L}_{loc}(d_i, g_{y_i})$$

cross-entropy classification loss

L_2 distance localization loss

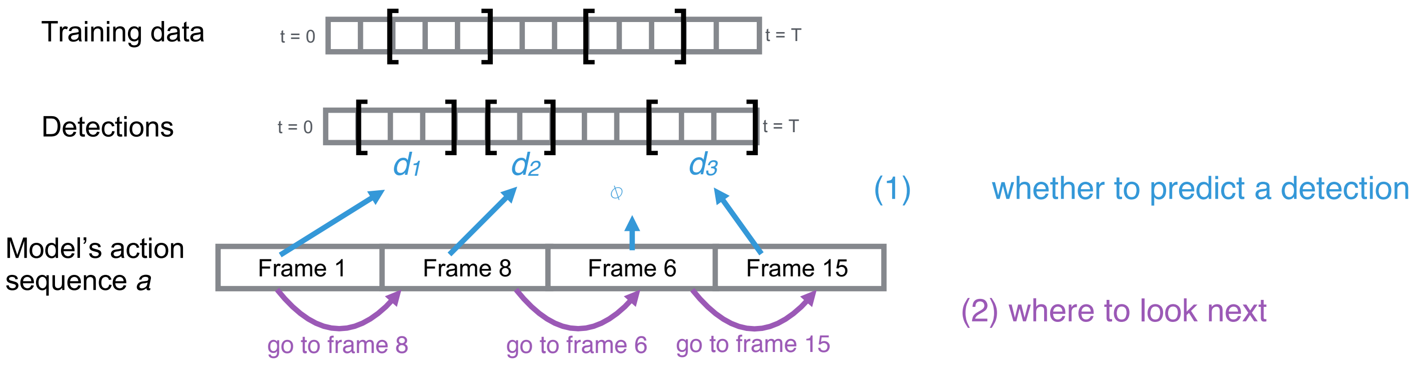
Yeung, Russakovsky, Mori, Fei-Fei. End-to-end Learning of Action Detection from Frame Glimpses in Videos. CVPR 2016.

Training the non-differentiable outputs



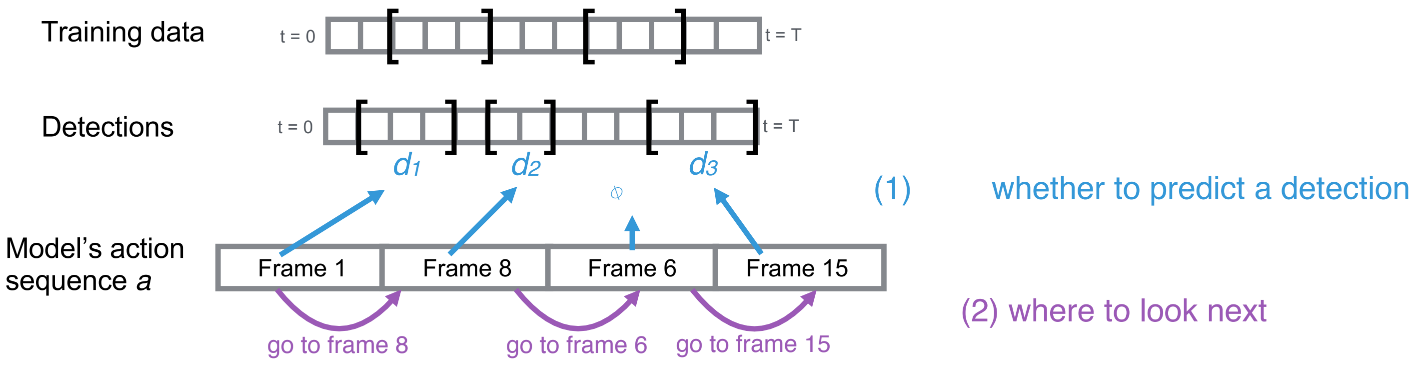
Yeung, Russakovsky, Mori, Fei-Fei. End-to-end Learning of Action Detection from Frame Glimpses in Videos. CVPR 2016.

Training the non-differentiable outputs



Yeung, Russakovsky, Mori, Fei-Fei. End-to-end Learning of Action Detection from Frame Glimpses in Videos. CVPR 2016.

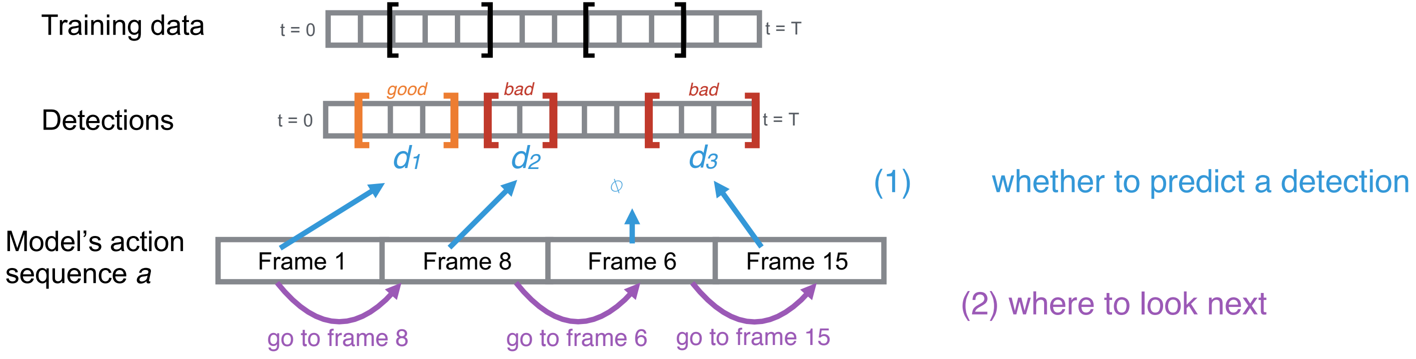
Training the non-differentiable outputs



Train an policy π_θ for actions (1) and (2) using REINFORCE [Williams 1992]

Yeung, Russakovsky, Mori, Fei-Fei. End-to-end Learning of Action Detection from Frame Glimpses in Videos. CVPR 2016.

Training the non-differentiable outputs

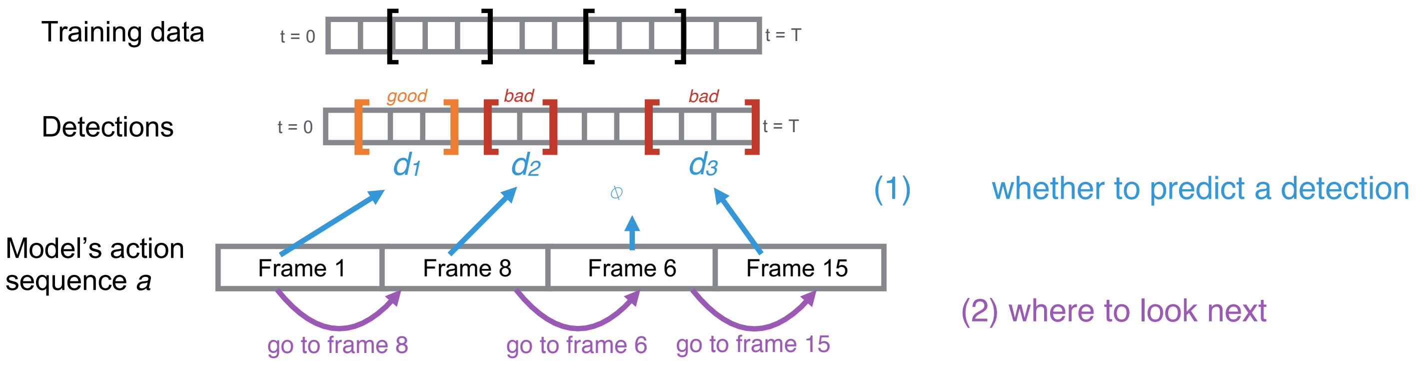


Train an policy π_θ for actions (1) and (2) using REINFORCE [Williams 1992]

Reward for an action sequence a: $$r(a) = \mathbf{N}^{+} - \alpha\mathbf{N}^{-}$$

Yeung, Russakovsky, Mori, Fei-Fei. End-to-end Learning of Action Detection from Frame Glimpses in Videos. CVPR 2016.

Training the non-differentiable outputs



Train an policy π_θ for actions (1) and (2) using REINFORCE [Williams 1992]

Reward for an action sequence a: $r(a) = \mathbf{N}^+ - \alpha \mathbf{N}^-$

Objective: $J(\theta) = \sum_a p_\theta(a) r(a)$

Gradient: $\nabla J(\theta) = \sum_a p_\theta(a) r(a) \nabla \log p_\theta(a)$

Monte-Carlo approximation: $\nabla J(\theta) \approx \frac{1}{K} \sum_{k=1}^{K} r(a^k) \sum_{t=1}^{T} \nabla \log \pi_\theta(a_t^k | M_t^k)$

Yeung, Russakovsky, Mori, Fei-Fei. End-to-end Learning of Action Detection from Frame Glimpses in Videos. CVPR 2016.

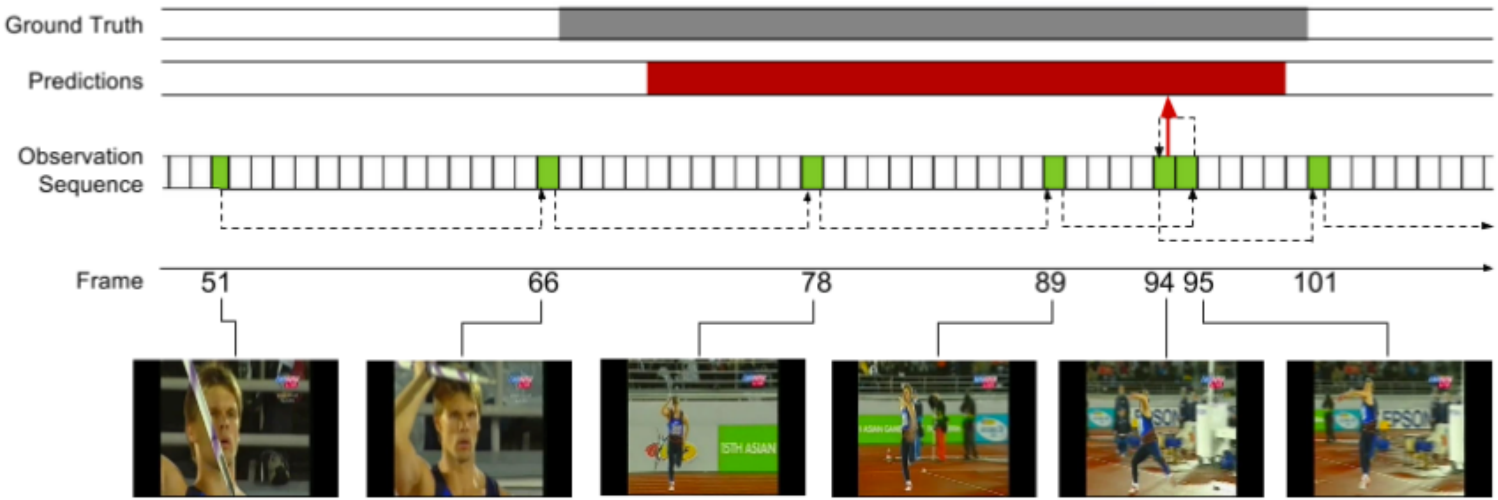
Action detection results

Dataset	Detection AP at IOU 0.5	
	State-of-the-art	Our result
THUMOS 2014	14.4	**17.1**
ActivityNet sports	33.2	**36.7**
ActivityNet work	31.1	**39.9**

While glimpsing only 2% of frames

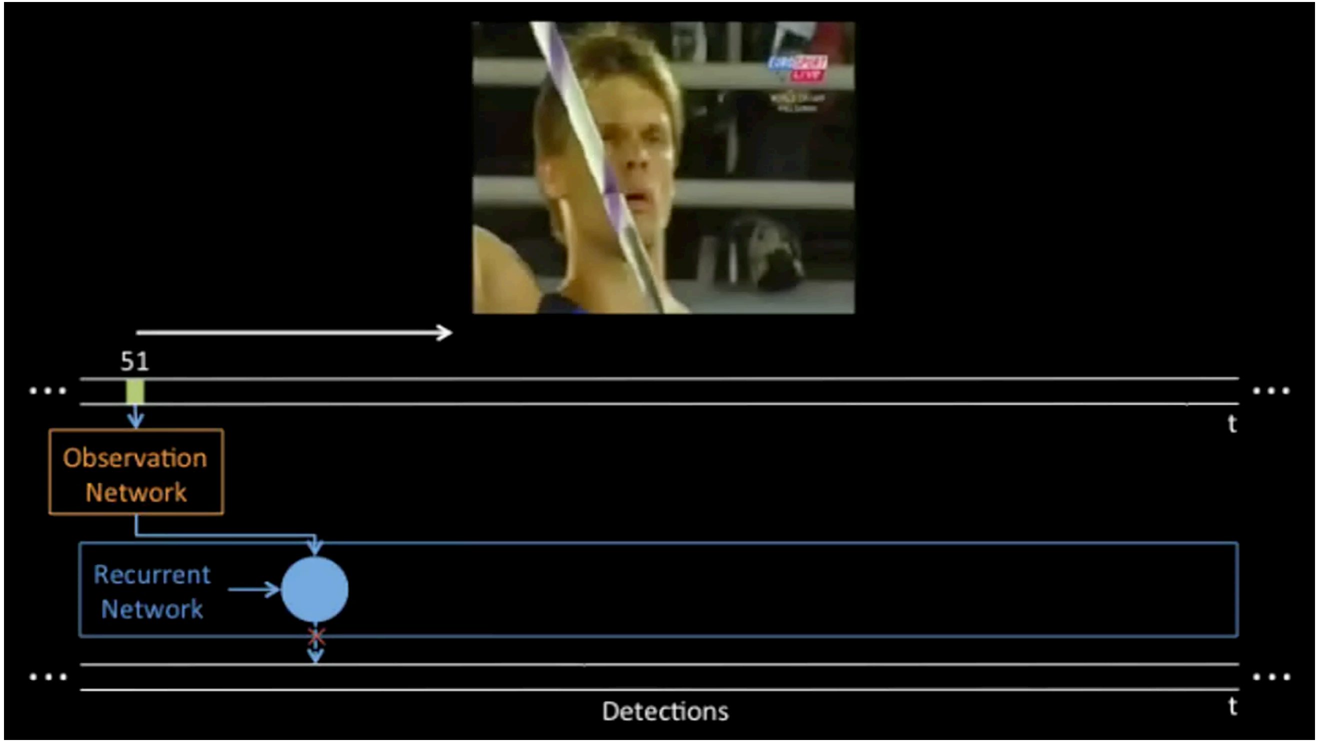
Yeung, Russakovsky, Mori, Fei-Fei. End-to-end Learning of Action Detection from Frame Glimpses in Videos. CVPR 2016.

Learned policies



Yeung, Russakovsky, Mori, Fei-Fei. End-to-end Learning of Action Detection from Frame Glimpses in Videos. CVPR 2016.

Learned policies



Yeung, Russakovsky, Mori, Fei-Fei. End-to-end Learning of Action Detection from Frame Glimpses in Videos. CVPR 2016.

Importance of prediction indicator output

	mAP (IOU = 0.5)
Ours (full model)	**17.1**
Ours w/o prediction indicator output (always predict)	**12.4**

Deciding when to output a prediction (learning to do non-maximum suppression) matters.

Yeung, Russakovsky, Mori, Fei-Fei. End-to-end Learning of Action Detection from Frame Glimpses in Videos. CVPR 2016.

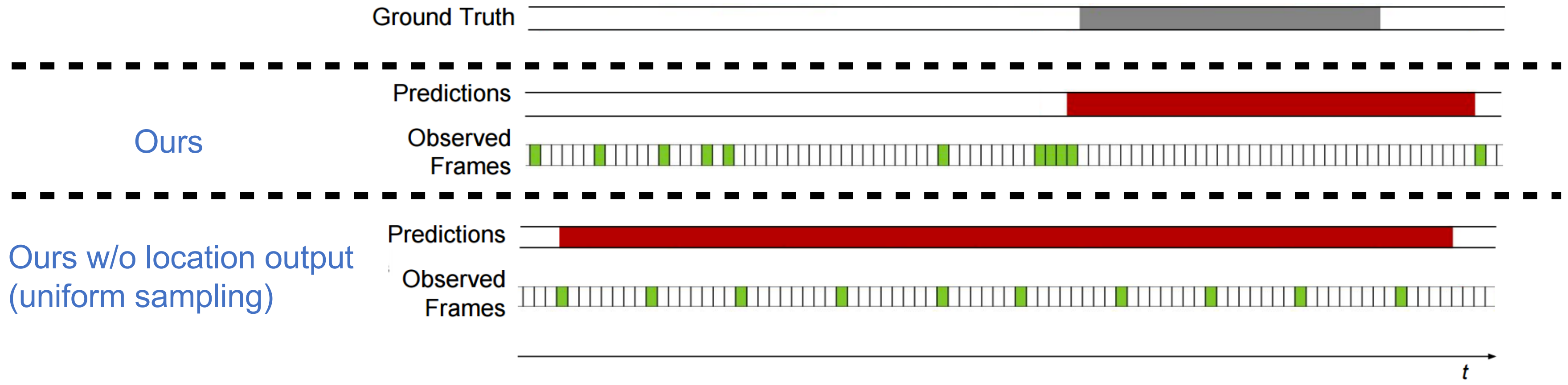
Importance of location output

	mAP (IOU = 0.5)
Ours (full model)	**17.1**
Ours w/o prediction indicator output (always predict)	12.4
Ours w/o location output (uniform sampling)	**9.3**

Deciding where to look next (location output) has even greater effect.

Yeung, Russakovsky, Mori, Fei-Fei. End-to-end Learning of Action Detection from Frame Glimpses in Videos. CVPR 2016.

Importance of location output



Uniform sampling does not always have sufficient temporal resolution where it's needed.

Yeung, Russakovsky, Mori, Fei-Fei. End-to-end Learning of Action Detection from Frame Glimpses in Videos. CVPR 2016.

Removing both prediction indicator and location outputs

	mAP (IOU = 0.5)
Ours (full model)	**17.1**
Ours w/o prediction indicator output (always predict)	12.4
Ours w/o location output (uniform sampling)	9.3
Ours w/o prediction indicator w/o location output (always predict, with uniform sampling)	**8.6**

Yeung, Russakovsky, Mori, Fei-Fei. End-to-end Learning of Action Detection from Frame Glimpses in Videos. CVPR 2016.

Importance of location regression

	mAP (IOU = 0.5)
Ours (full model)	**17.1**
Ours w/o prediction indicator output (always predict)	12.4
Ours w/o location output (uniform sampling)	9.3
Ours w/o prediction indicator w/o location output (always predict, with uniform sampling)	8.6
Ours w/o location regression (always output mean action duration)	**5.5**

Simply outputting mean action duration gives significantly worse performance.

Yeung, Russakovsky, Mori, Fei-Fei. End-to-end Learning of Action Detection from Frame Glimpses in Videos. CVPR 2016.

Desiderata for Activity Recognition Models

Label structure

Hu et al., CVPR 16
Deng et al., CVPR 16
Nauata et al., CVPRW 17
Deng et al., CVPR 17

Temporal structure

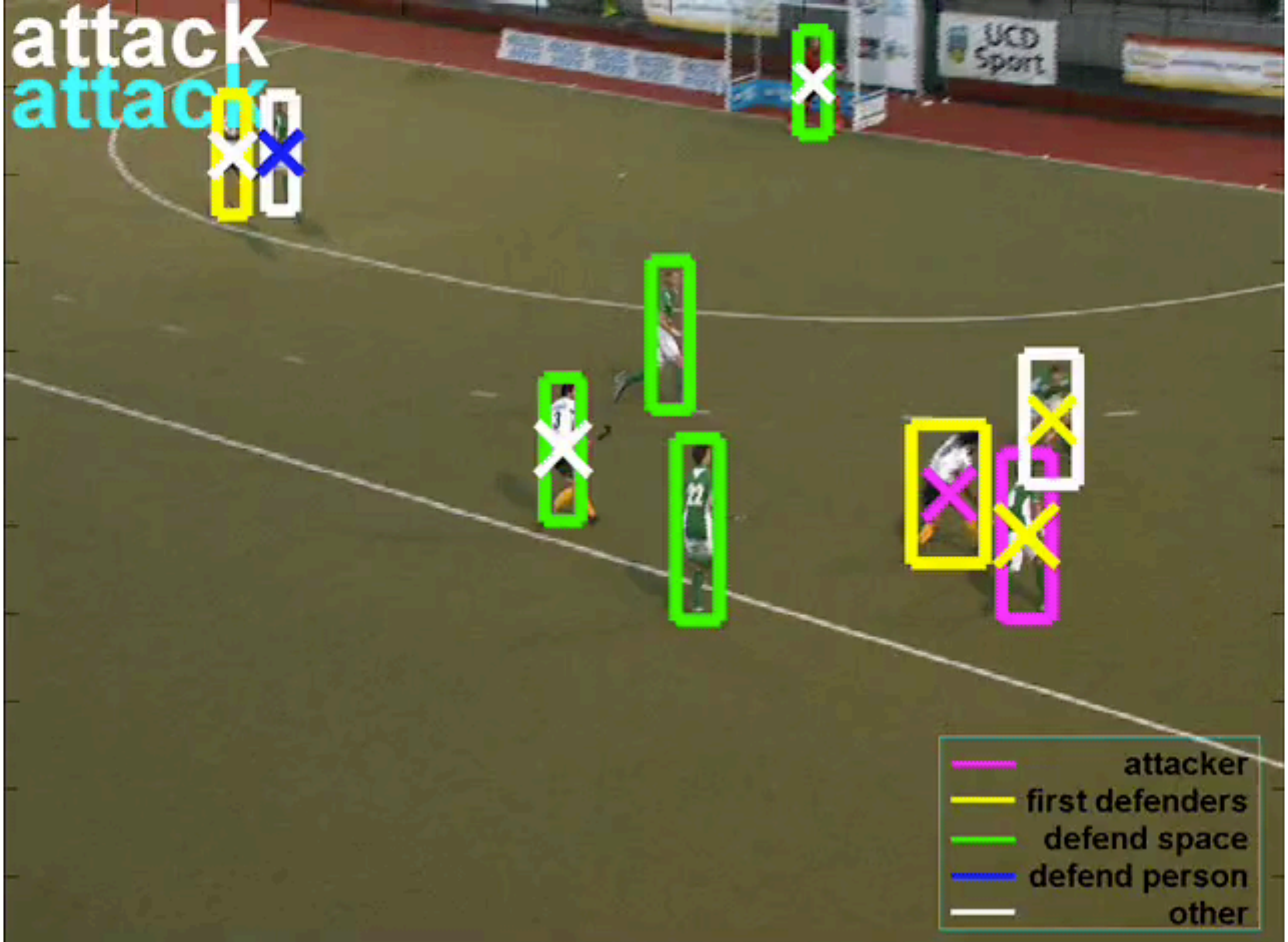
Yeung et al., CVPR 16
Yeung et al., IJCV 17
He et al., WACV 18
Chen et al., ICCVW 17

Group structure

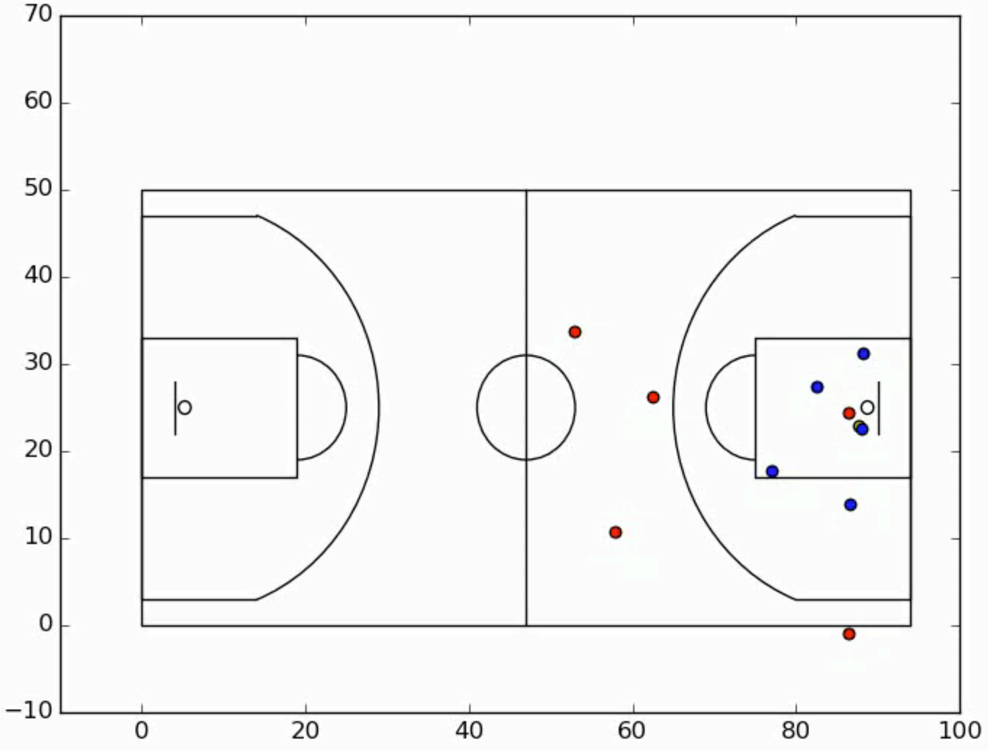
Ibrahim et al., CVPR 16
Mehrasa et al., arXiv 17
Khodabandeh et al., arXiv 17
Lan et al. CVPR 12

Role of Context in Actions

Who has the puck?


attack
attack
UCD Sport
attacker
first defenders
defend space
defend person
other

Analyzing Human Trajectories to Recognize Actions



Which team is it?

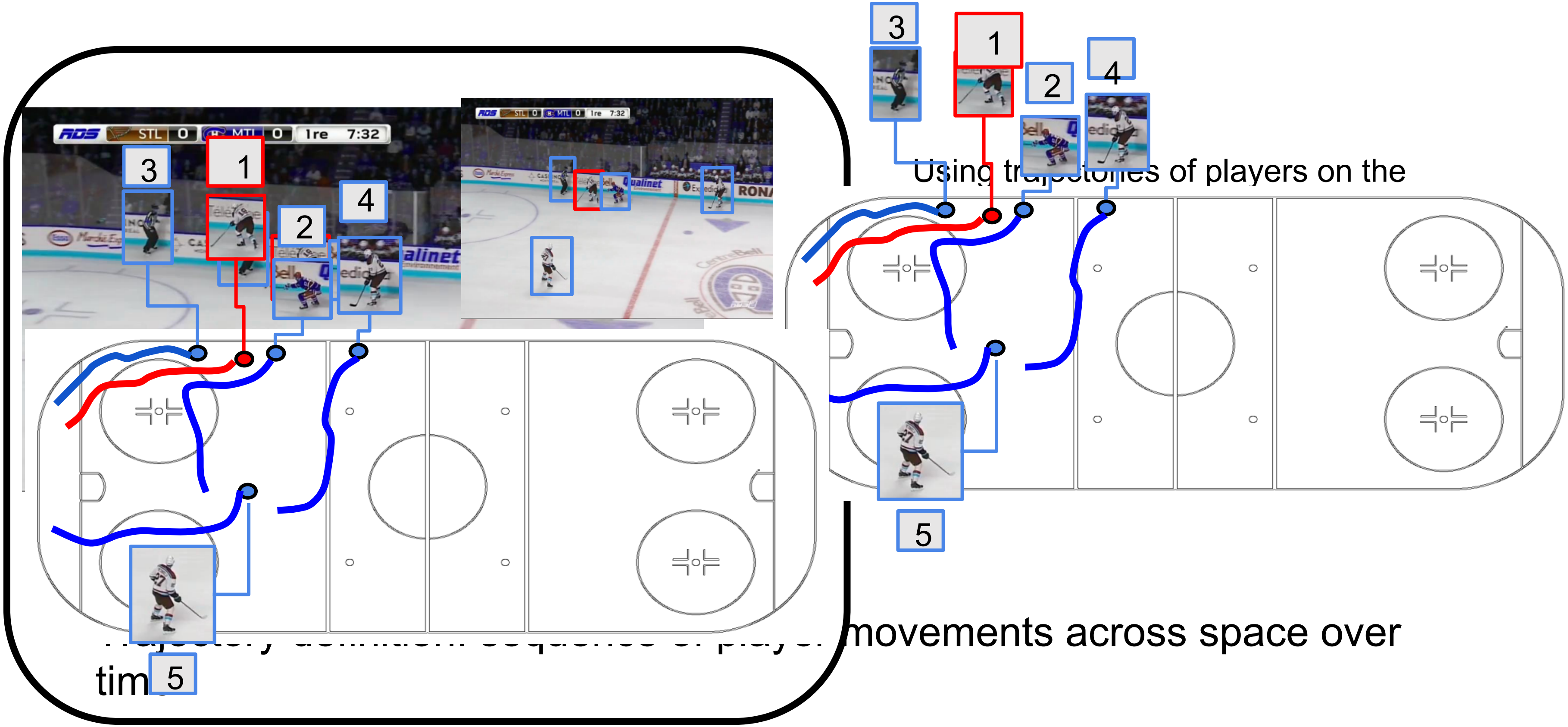
Who was player X?

Will the shot be successful?

Mehrasa, Zhong, Tung, Bornn, Mori, Learning Person Trajectory Representations for Team Activity Analysis, arXiv 2017

Motivation

Using tr… …ories of players on the

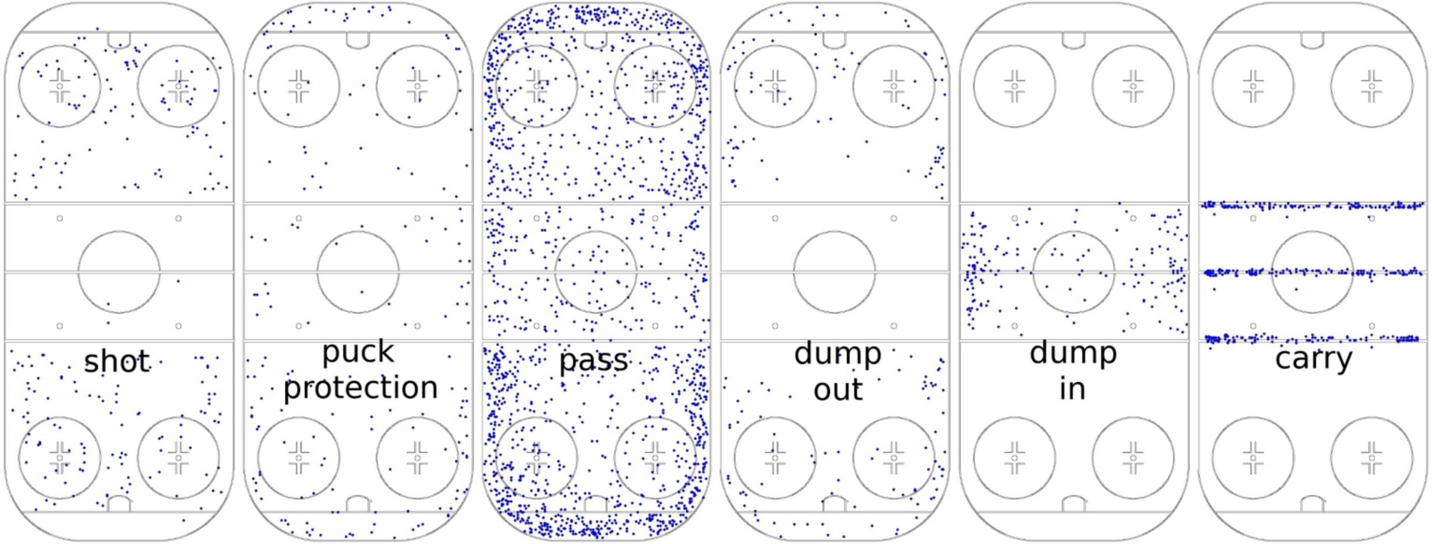


… movements across space over

tim…

Mehrasa, Zhong, Tung, Bornn, Mori, Learning Person Trajectory Representations for Team Activity Analysis, arXiv 2017

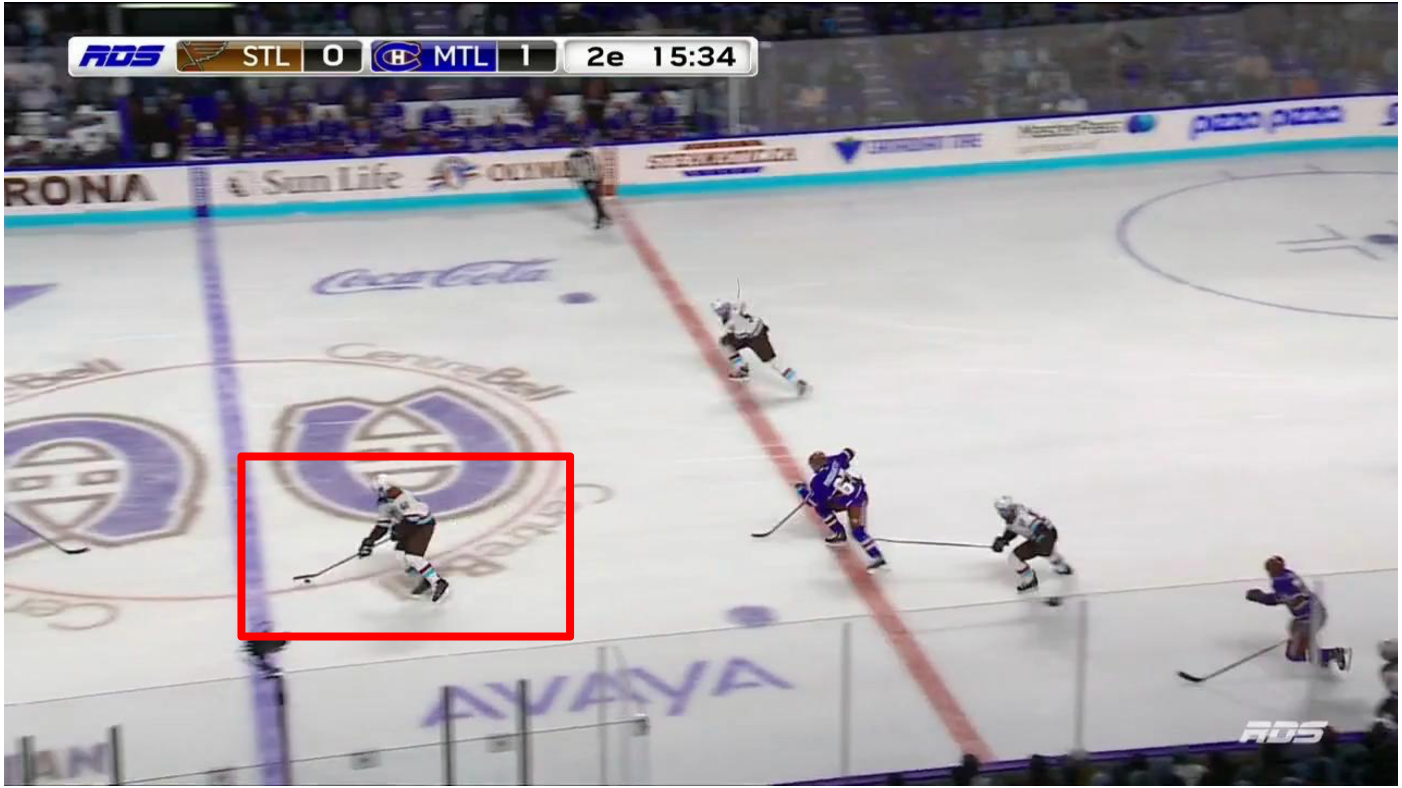
Motivation



locations matter!

Mehrasa, Zhong, Tung, Bornn, Mori, Learning Person Trajectory Representations for Team Activity Analysis, arXiv 2017

Key Player Definition



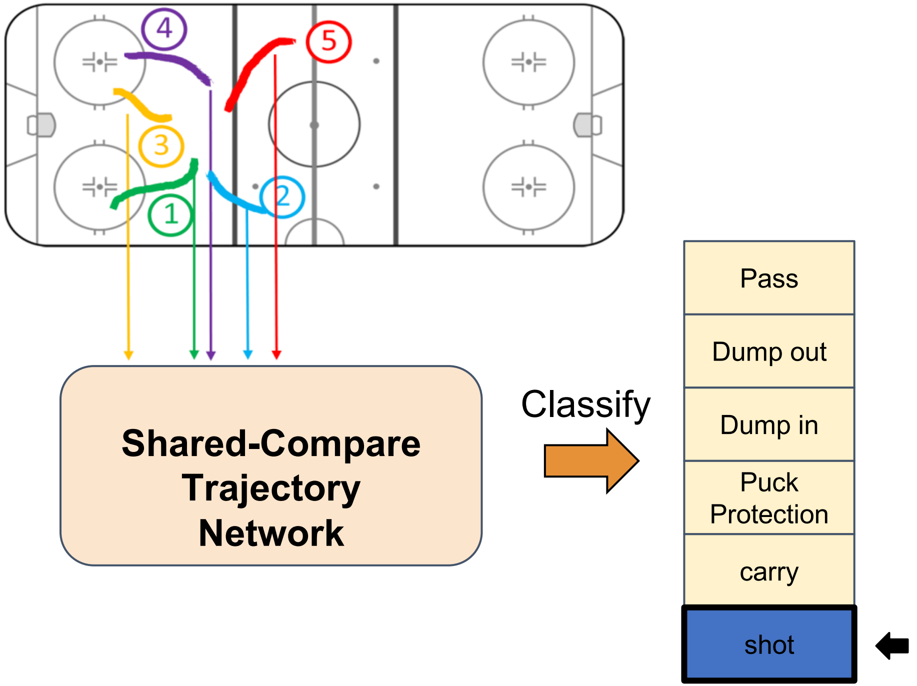
Mehrasa, Zhong, Tung, Bornn, Mori, Learning Person Trajectory Representations for Team Activity Analysis, arXiv 2017

Model and Approach

- **Shared-Compare Trajectory Network**
- Stacked Trajectory Network

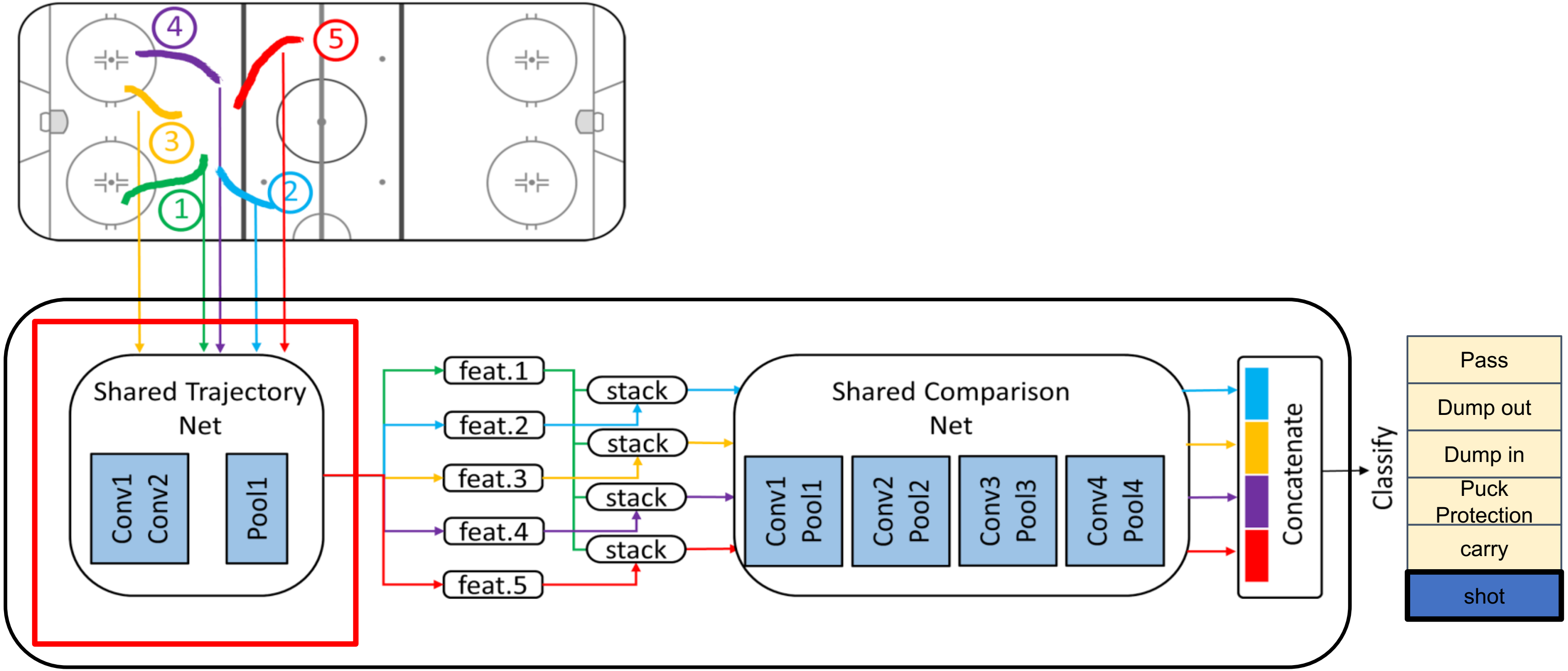
Mehrasa, Zhong, Tung, Bornn, Mori, Learning Person Trajectory Representations for Team Activity Analysis, arXiv 2017

Shared-Compare Trajectory Network



Mehrasa, Zhong, Tung, Bornn, Mori, Learning Person Trajectory Representations for Team Activity Analysis, arXiv 2017

Shared-Compare Trajectory Network

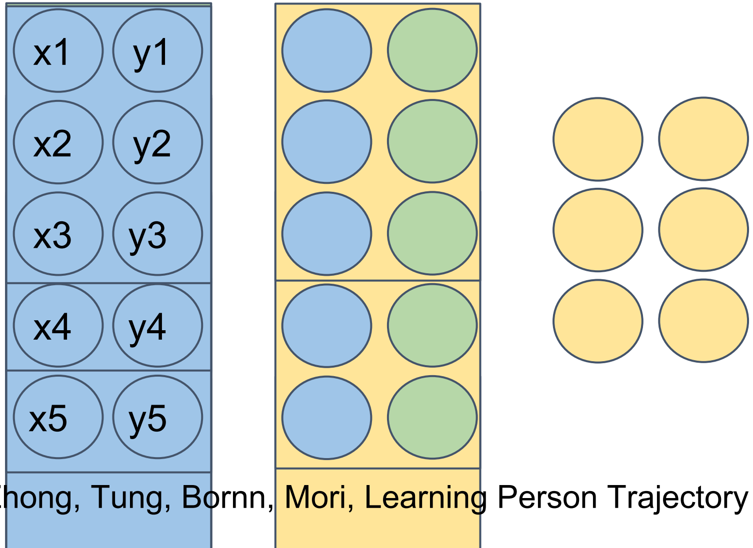


Shared-Compare Trajectory Network

Shared Trajectory Network

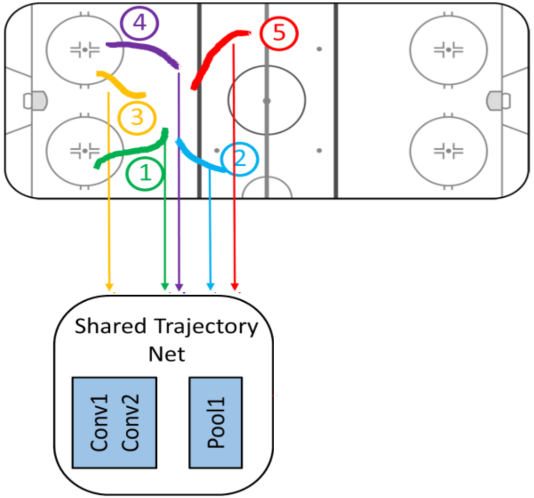
- Consists of 1D convolution and max-pooling
- Learning generic representation for each ind

1D max-pooling layer



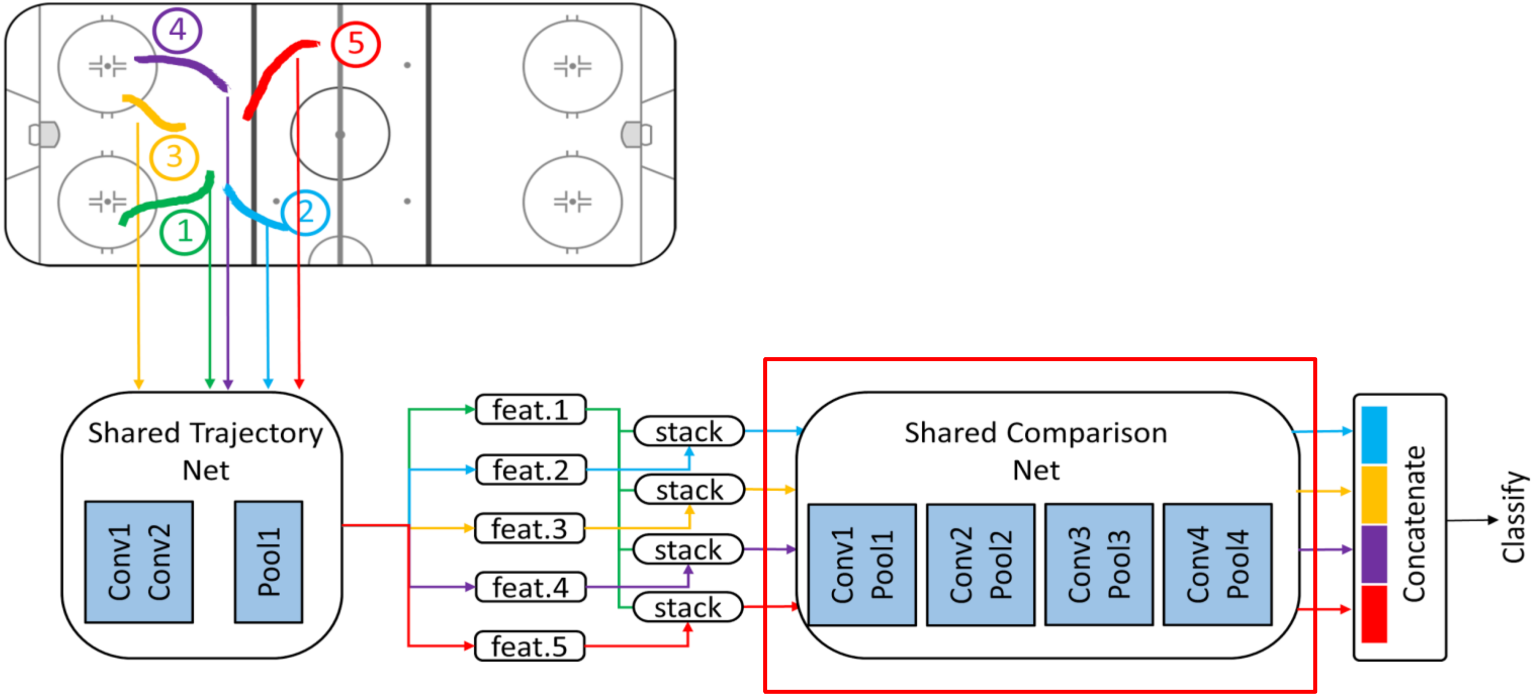
Kernel Size =C * K * M

Pooling stride =2



Mehrasa, Zhong, Tung, Bornn, Mori, Learning Person Trajectory Representations for Team Activity Analysis, arXiv 2017

Shared-Compare Trajectory Network

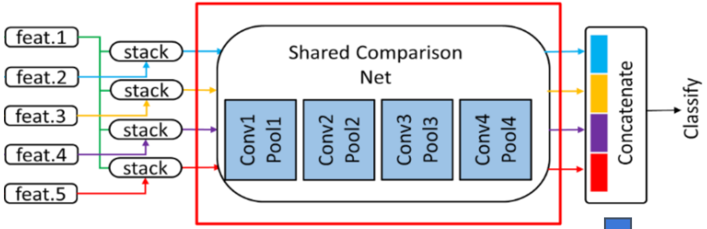


Mehrasa, Zhong, Tung, Bornn, Mori, Learning Person Trajectory Representations for Team Activity Analysis, arXiv 2017

Shared Compare Network

Input:

- Pairs of individual trajectory features provided by Shared Trajectory Network
- Pairs are formed relative to a “key player”



Enforce an ordering among the players

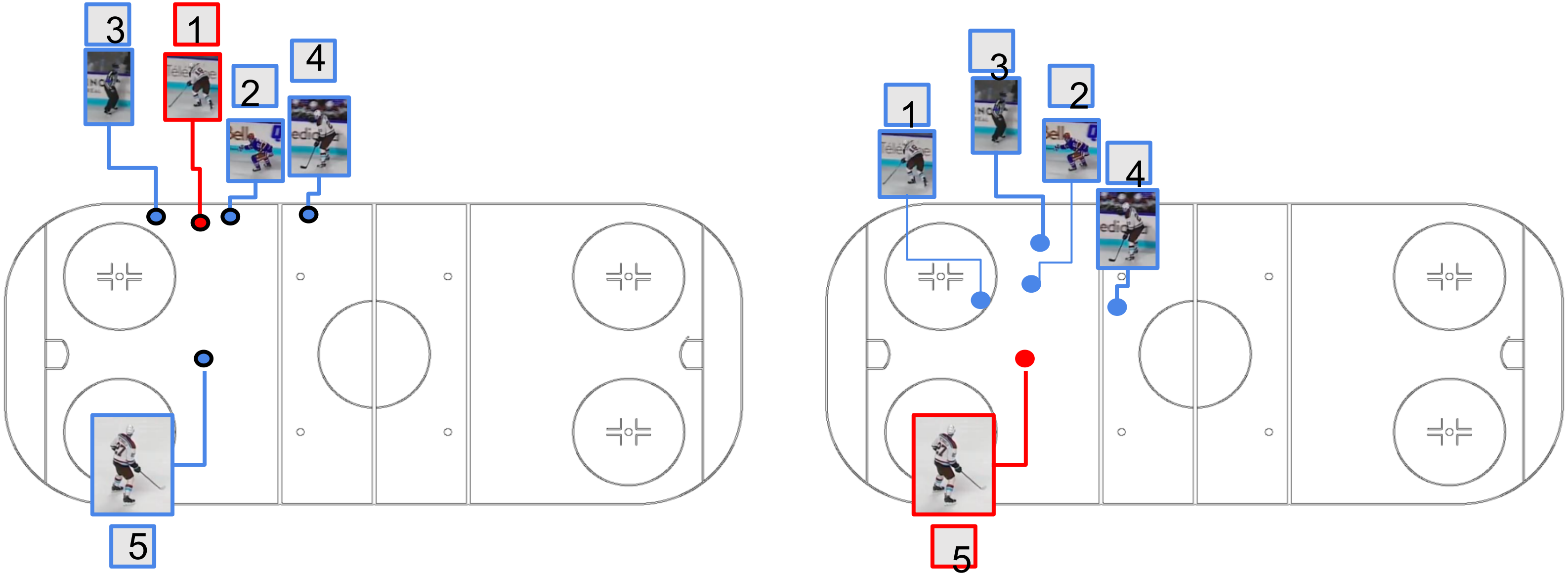
Learning:

- The relative motion patterns of pairs
- Interaction cues of players

Output: relative motion pattern representation of each pair

Mehrasa, Zhong, Tung, Bornn, Mori, Learning Person Trajectory Representations for Team Activity Analysis, arXiv 2017

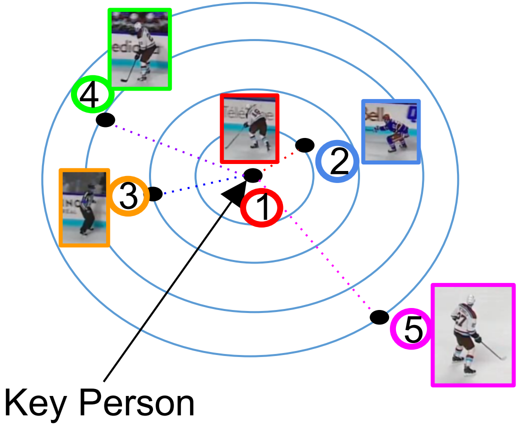
Players Ordering

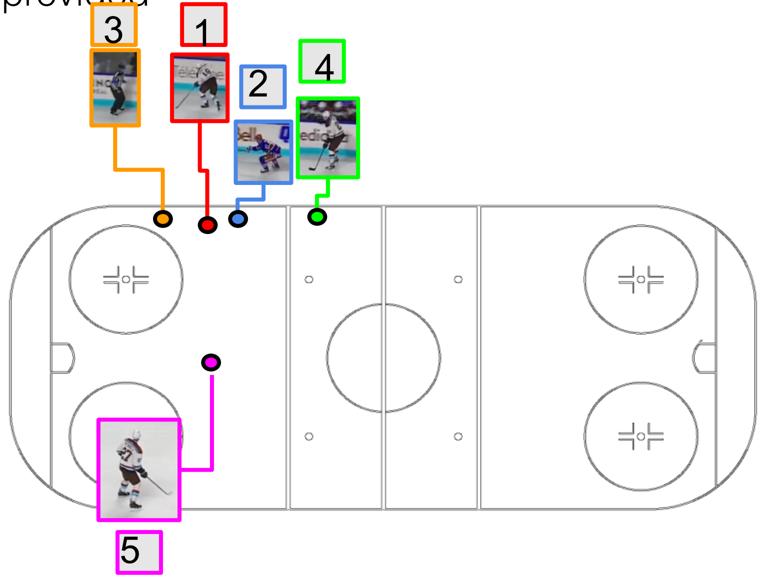


Mehrasa, Zhong, Tung, Bornn, Mori, Learning Person Trajectory Representations for Team Activity Analysis, arXiv 2017

Relative Ordering

- Spatial proximity to the key player
- Key person may not be available in a general non-sports setting
- Average pooling strategy when key player is not provided





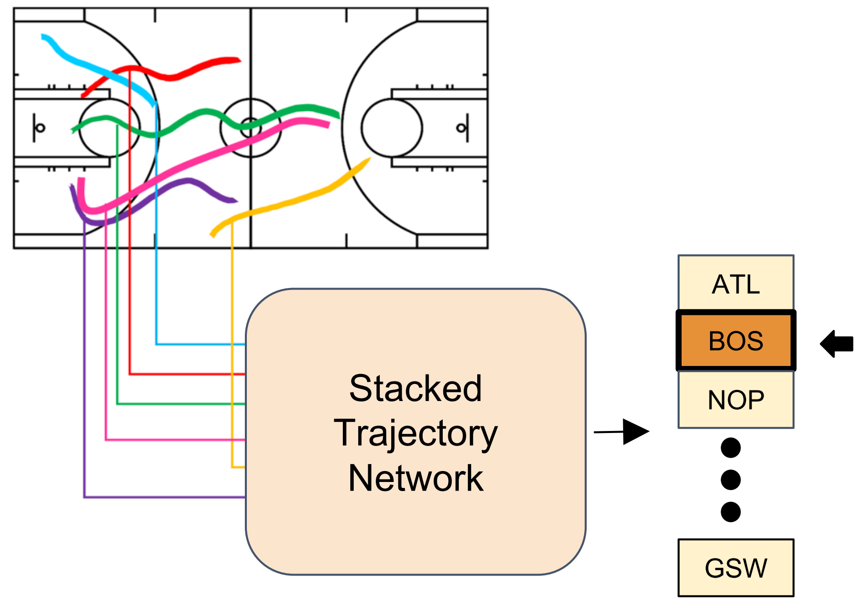
Mehrasa, Zhong, Tung, Bornn, Mori, Learning Person Trajectory Representations for Team Activity Analysis, arXiv 2017

Model and Approach

- Shared-Compare Trajectory Network
- **Stacked Trajectory Network**

Mehrasa, Zhong, Tung, Bornn, Mori, Learning Person Trajectory Representations for Team Activity Analysis, arXiv 2017

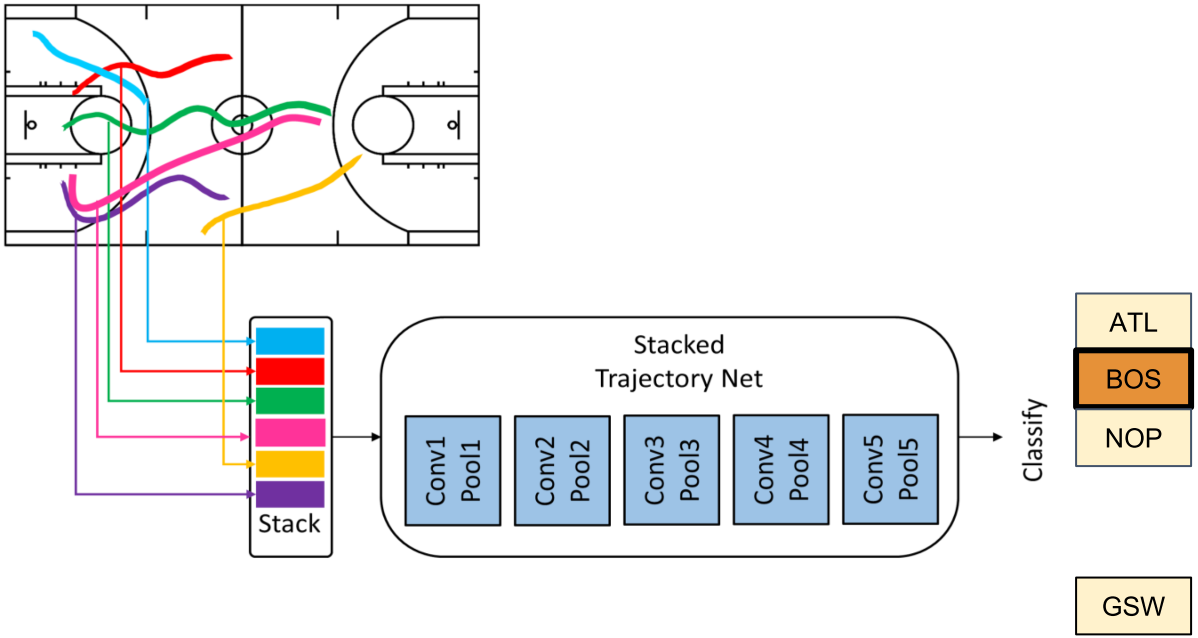
Stacked Trajectory Network



Mehrasa, Zhong, Tung, Bornn, Mori, Learning Person Trajectory Representations for Team Activity Analysis, arXiv 2017

Stacked Trajectory Network

- Learning overall group dynamics



Mehrasa, Zhong, Tung, Bornn, Mori, Learning Person Trajectory Representations for Team Activity Analysis, arXiv 2017

Experiments

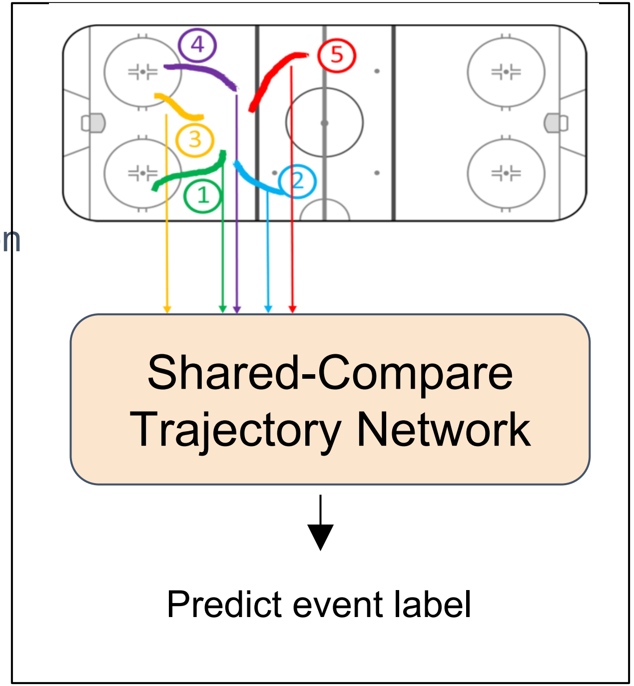
- **Event Recognition on the Sportlogiq Dataset**
- Team Identification on the NBA Dataset

Mehrasa, Zhong, Tung, Bornn, Mori, Learning Person Trajectory Representations for Team Activity Analysis, arXiv 2017

Event recognition using Sportlogiq dataset

Task Definition

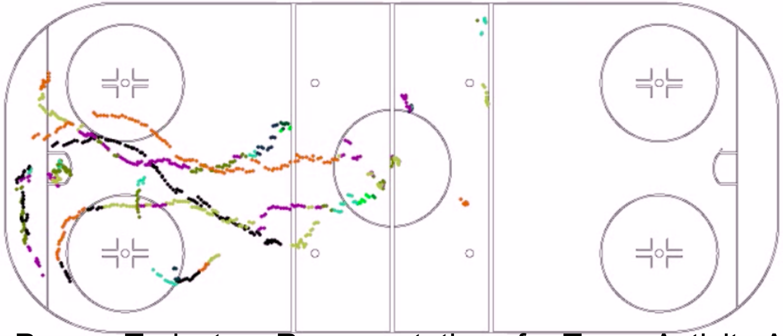
- Event classification
- 6 event classes
  - pass, dump in, dump out, shot, carry, puck protection
- Dataset: Sportlogiq hockey dataset



Mehrasa, Zhong, Tung, Bornn, Mori, Learning Person Trajectory Representations for Team Activity Analysis, arXiv 2017

Event recognition using Sportlogiq dataset

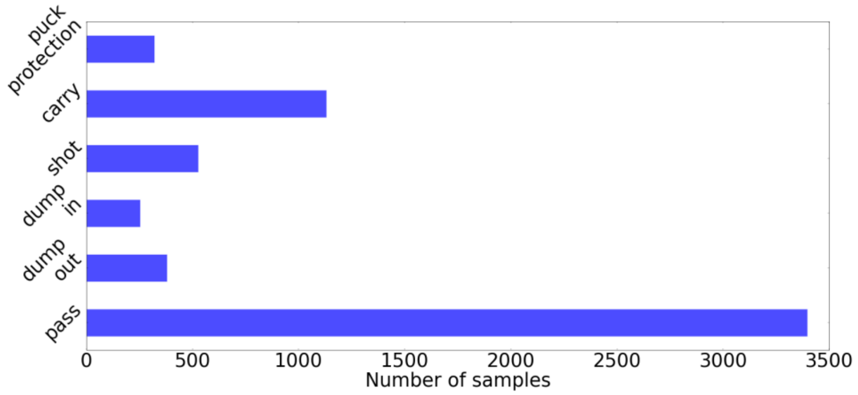
How the Sportlogiq dataset looks



Mehrasa, Zhong, Tung, Bornn, Mori, Learning Person Trajectory Representations for Team Activity Analysis, arXiv 2017

Event recognition using Sportlogiq dataset

- Sportlogiq Dataset Information
  - State of the art algorithms are used to automatically detect and track players in raw broadcast video
  - Trajectory data are estimated using homography
  - Trajectory length: 16 frames
  - # players used is fixed: 5
  - # of samples of each event



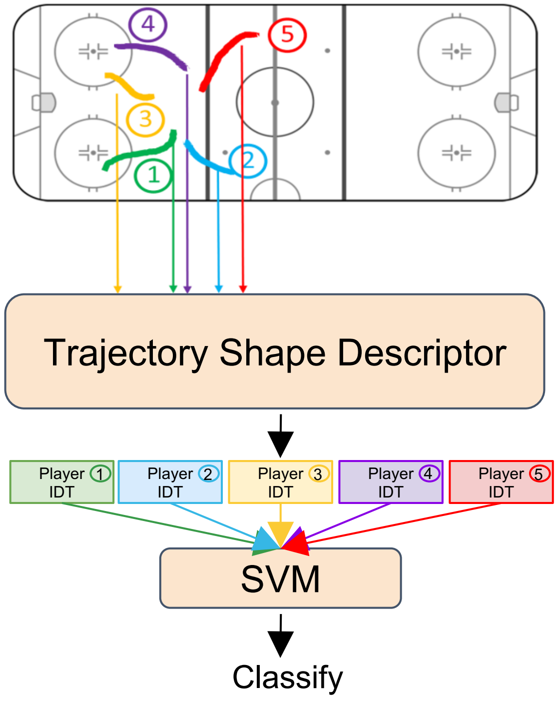
  - 4 games for training, 2 games for validation, and 2 games for testing

Mehrasa, Zhong, Tung, Bornn, Mori, Learning Person Trajectory Representations for Team Activity Analysis, arXiv 2017

Event recognition using Sportlogiq dataset

Baselines:

- IDT[1]
  - Same input data as in our method
  - Each trajectory as IDT *Trajectory shape descriptor*
  - Normalized displacement vector of trajectory
  - SVM with RBF kernel and 'one vs. rest' mechanism

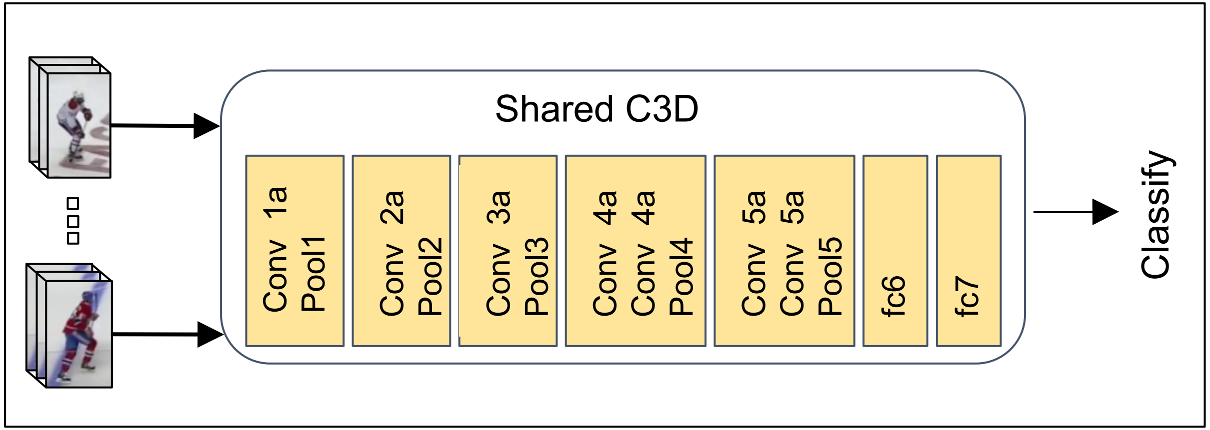


[1] Wang et al., Dense trajectories and motion boundary descriptors for action recognition. IJCV 2013.

Event recognition using Sportlogiq dataset

Baselines:

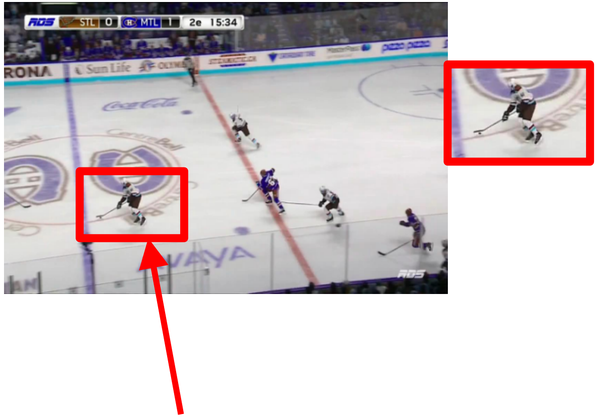
- C3D[1]
    - Trained from scratch
    - Fine-tuned from a model pretrained on Sports-1M
    - Same ordering as in our approach



[1] Tran et al., Learning spatiotemporal features with 3d convolutional networks. CVPR 2015.

Event recognition using Sportlogiq dataset

- Training phase:
  - Key player is provided
  - Remaining players are ranked by proximity to the key player
- Test phase:
  - Both cases of known and unknown key player
  - Average pooling strategy for the case of unknown key player



Key Player

Event recognition on Sportlogiq dataset

Unknown Key Player

	IDT	C3D	Fine-tuned C3D	Shared-Cmp
pass	72.86%	71.10%	77.45%	78.13%
dump out	13.75%	11.66%	18.15%	22.14%
dump in	6.35%	7.58%	19.04%	26.63%
shot	13.05%	23.37%	38.96%	40.52%
carry	45.66%	64.75%	65.65%	61.10%
puck protection	6.28%	6.50%	7.98%	8.72%
mAP	26.32%	30.83%	37.87%	39.54%

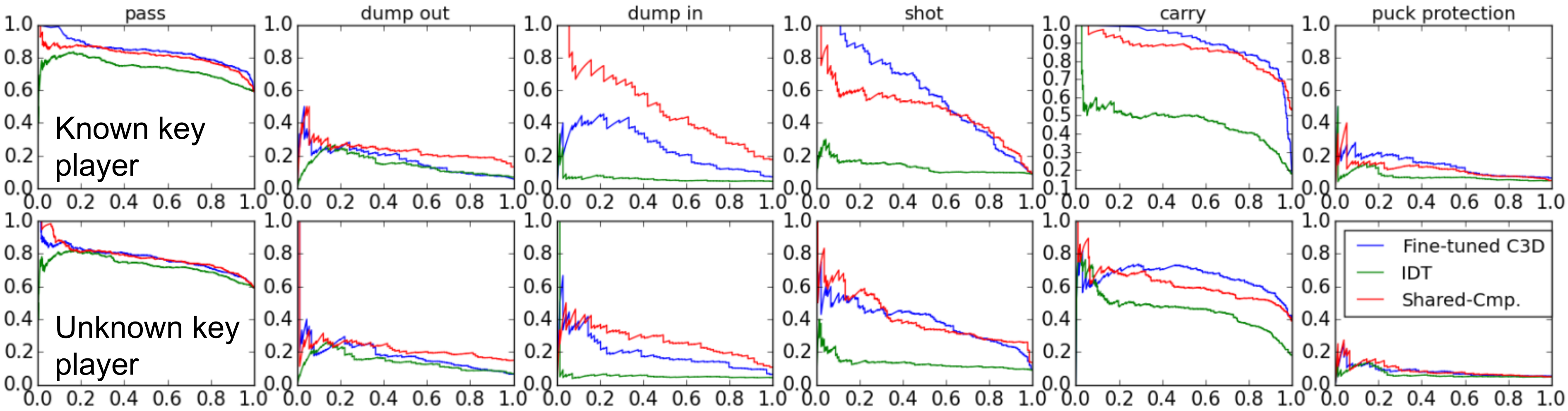
Known Key Player

	IDT	C3D	Fine-tuned C3D	Shared-Cmp
pass	73.35%	77.30%	84.34%	81.33%
dump out	14.34%	10.17%	17.10%	23.11%
dump in	5.77%	10.25%	24.83%	50.04%
shot	13.07%	34.17%	58.88%	48.51%
carry	47.38%	86.37%	90.10%	85.96%
puck protection	7.28%	11.83%	13.99%	11.54%
mAP	26.86%	38.35%	48.21%	50.08%

- In comparison to IDT 13.2 higher mAP
- In comparison to C3D trained from scratch 8.7 higher mAP
- In comparison to fine-tuned C3D 1.7 higher mAP

Event recognition on Sportlogiq dataset

Precision-recall curve



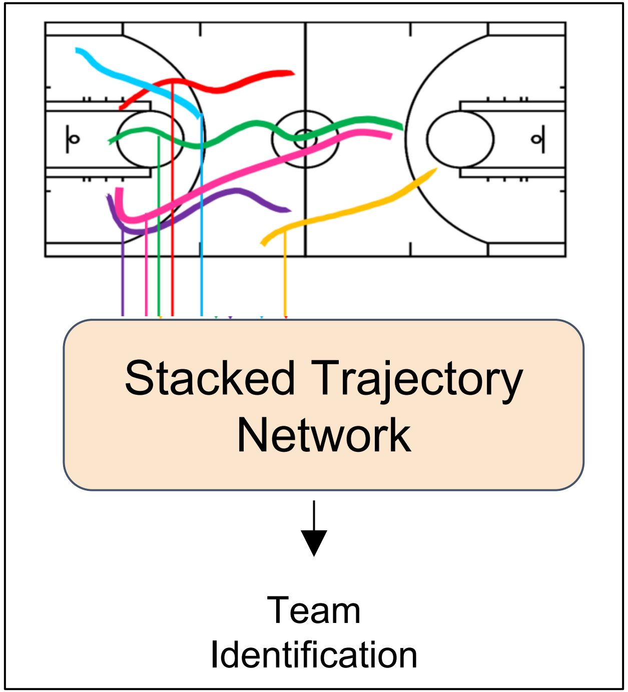
Experiments

- Event Recognition on the Sportlogiq Dataset
- **Team Identification on the NBA Dataset**

Team Identification on the NBA Dataset

Task Definition

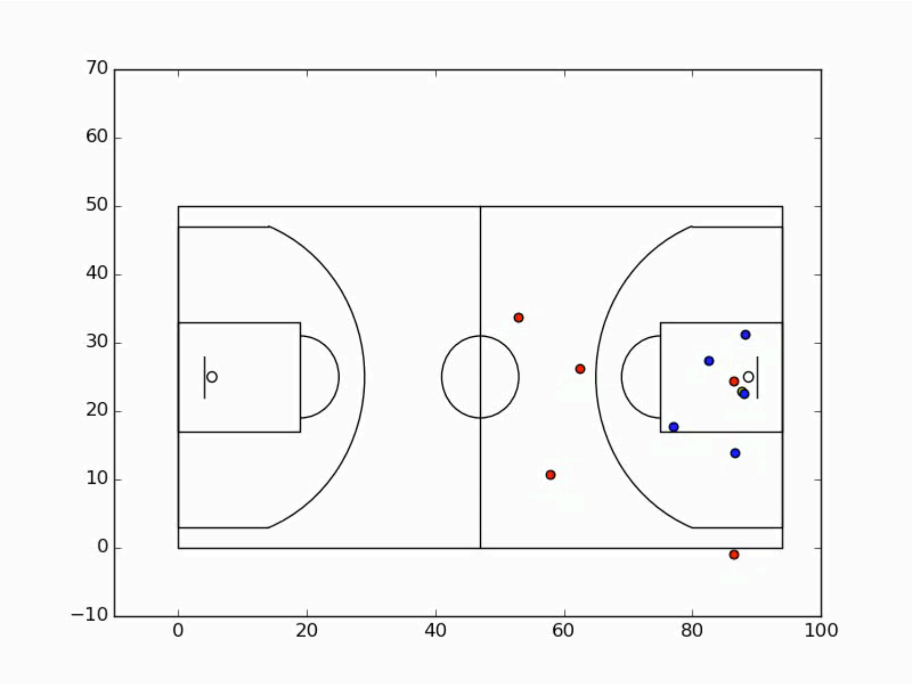
- Team Identification
- Stacked Trajectory Network
- 30 NBA teams
- Dataset: NBA basketball dataset



Mehrasa, Zhong, Tung, Bornn, Mori, Learning Person Trajectory Representations for Team Activity Analysis, arXiv 2017

Team Identification on the NBA Dataset

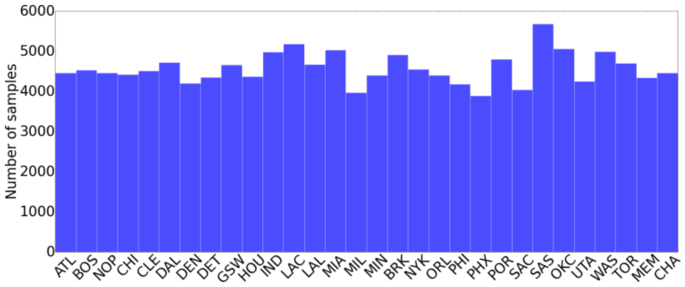
How the NBA dataset looks like



Mehrasa, Zhong, Tung, Bornn, Mori, Learning Person Trajectory Representations for Team Activity Analysis, arXiv 2017

Team Identification using NBA dataset

- Dataset Information
  - Trajectory data are acquired by a multi-camera system
  - Sampling rate: 25Hz
  - Extract 137176 possessions from 1076 games
  - 200 frames per possession
  - 82375 poss. for training, 27437 poss. for testing, and 27437 poss. for validation
  - Number of poss. per team



Mehrasa, Zhong, Tung, Bornn, Mori, Learning Person Trajectory Representations for Team Activity Analysis, arXiv 2017

Team Identification on the NBA Dataset

Results

layers	acc	hit@2	hit@3	game acc
2conv	10.68%	18.09%	24.31%	50.00%
3conv	18.86%	28.89%	36.47%	87.05%
4conv	22.34%	33.03%	40.47%	93.41%
5conv	24.78%	35.61%	42.95%	95.91%
5conv+2fc	25.08%	35.83%	42.85%	94.32%

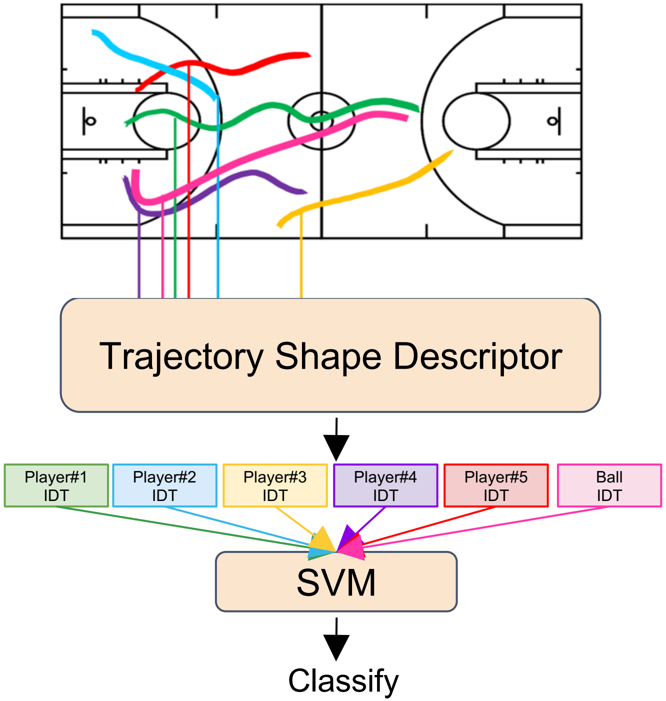
Mehrasa, Zhong, Tung, Bornn, Mori, Learning Person Trajectory Representations for Team Activity Analysis, arXiv 2017

Team Identification on the NBA Dataset

Baseline:

- IDT[1]
  - Same input data as in our method
  - Each trajectory as IDT Trajectory shape descriptor
  - SVM with RBF kernel and ‘one vs. rest’ mechanism

models	acc	game acc
IDT	5.74%	9.10%
Stacked Traj. Net	25.78%	95.91%



[1] Wang et al., Dense trajectories and motion boundary descriptors for action recognition. IJCV 2013.

Summary

- Learning person trajectory representations for group activity analysis
- Using deep neural network models for learning trajectory features

- Experiments shows our model is capable of capturing:
    - Complex spatial-temporal dependencies
    - Distinctive group dynamics

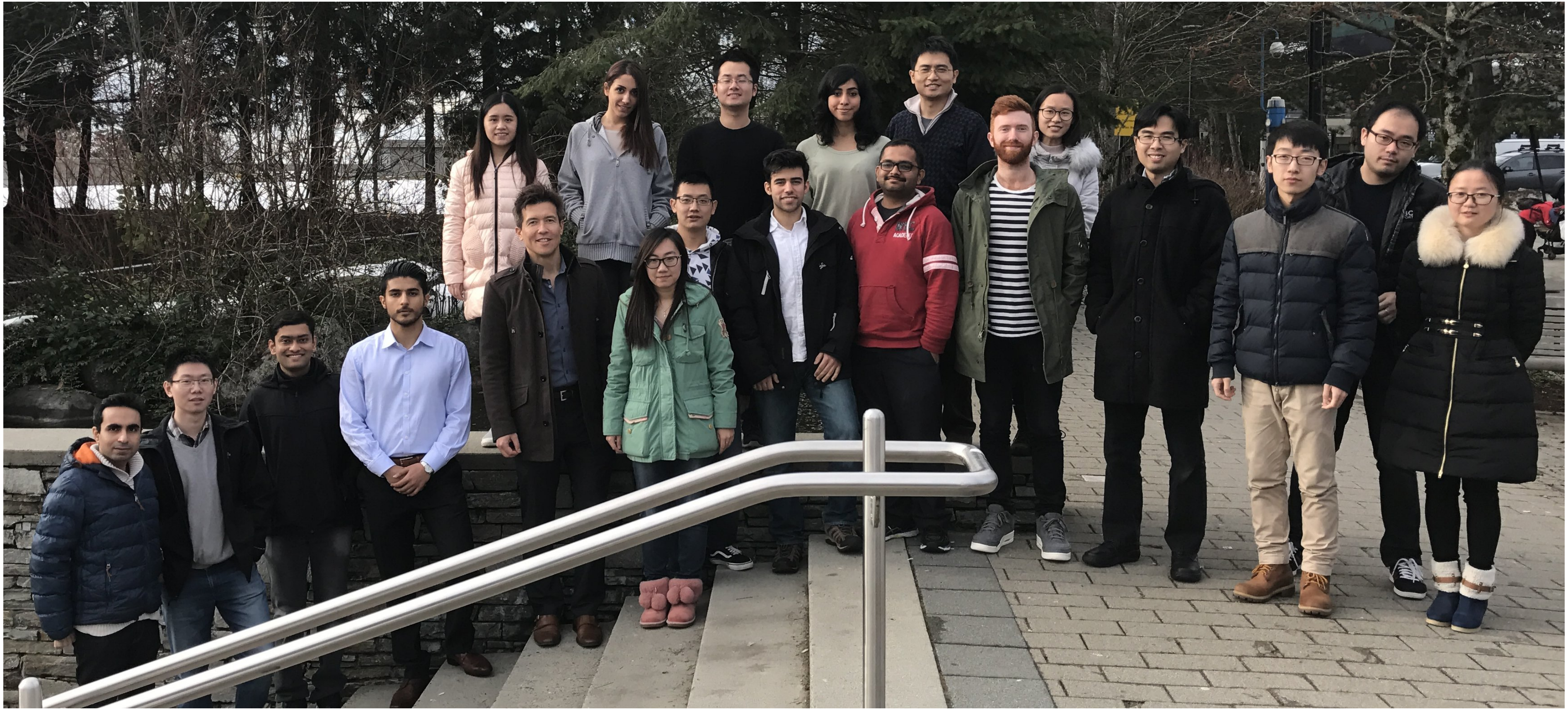
Conclusion

Methods for handling *structures* in deep networks

Label structure: message passing algorithms for multi-level image/video labeling; purely from image data or with partial labels

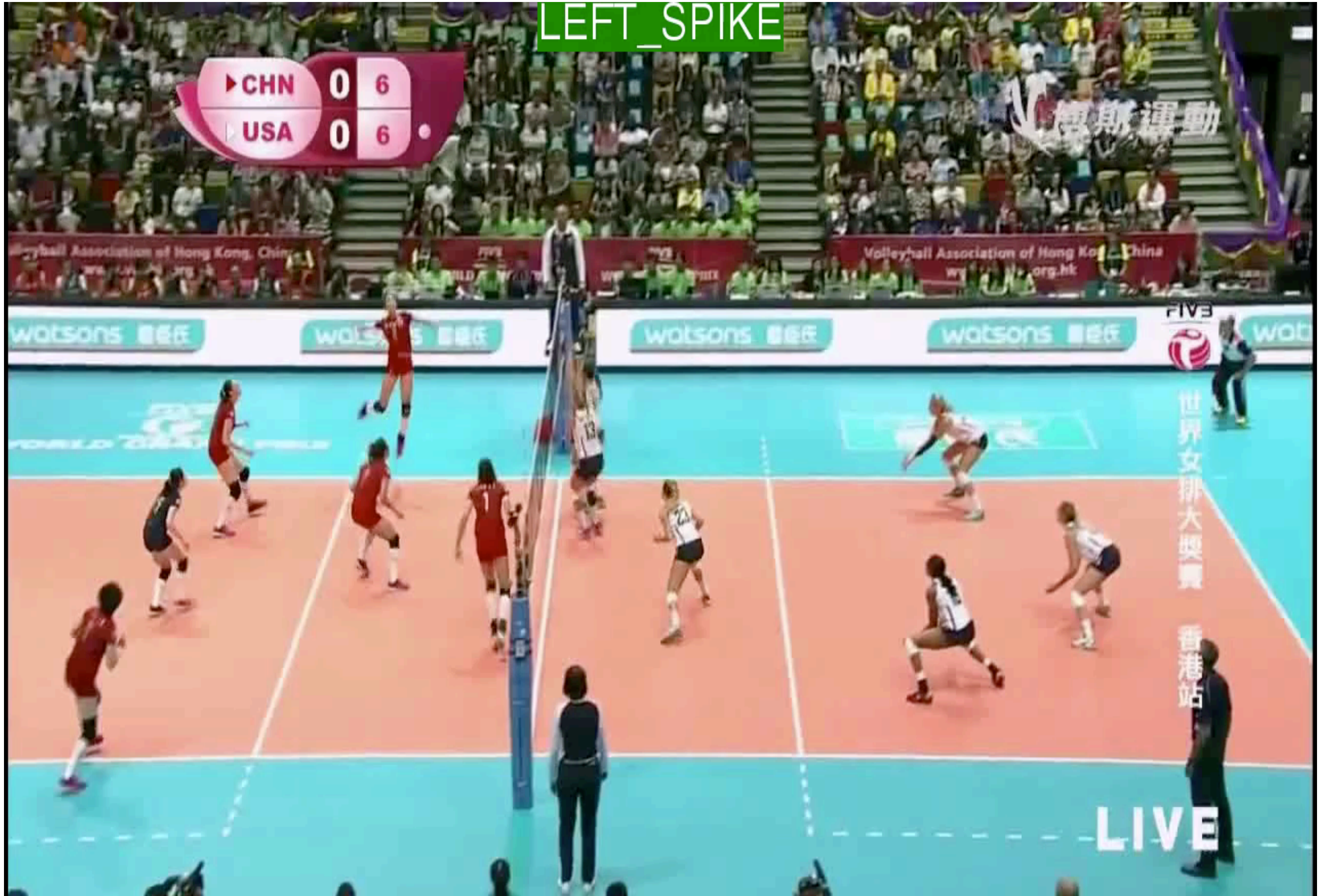
Temporal structure: action detection in time; efficient glimpsing of video frames

Group structure: network structures to connect related people, gating functions or modules for reasoning about relations



Thank you!

Example: Rally in a Volleyball Game



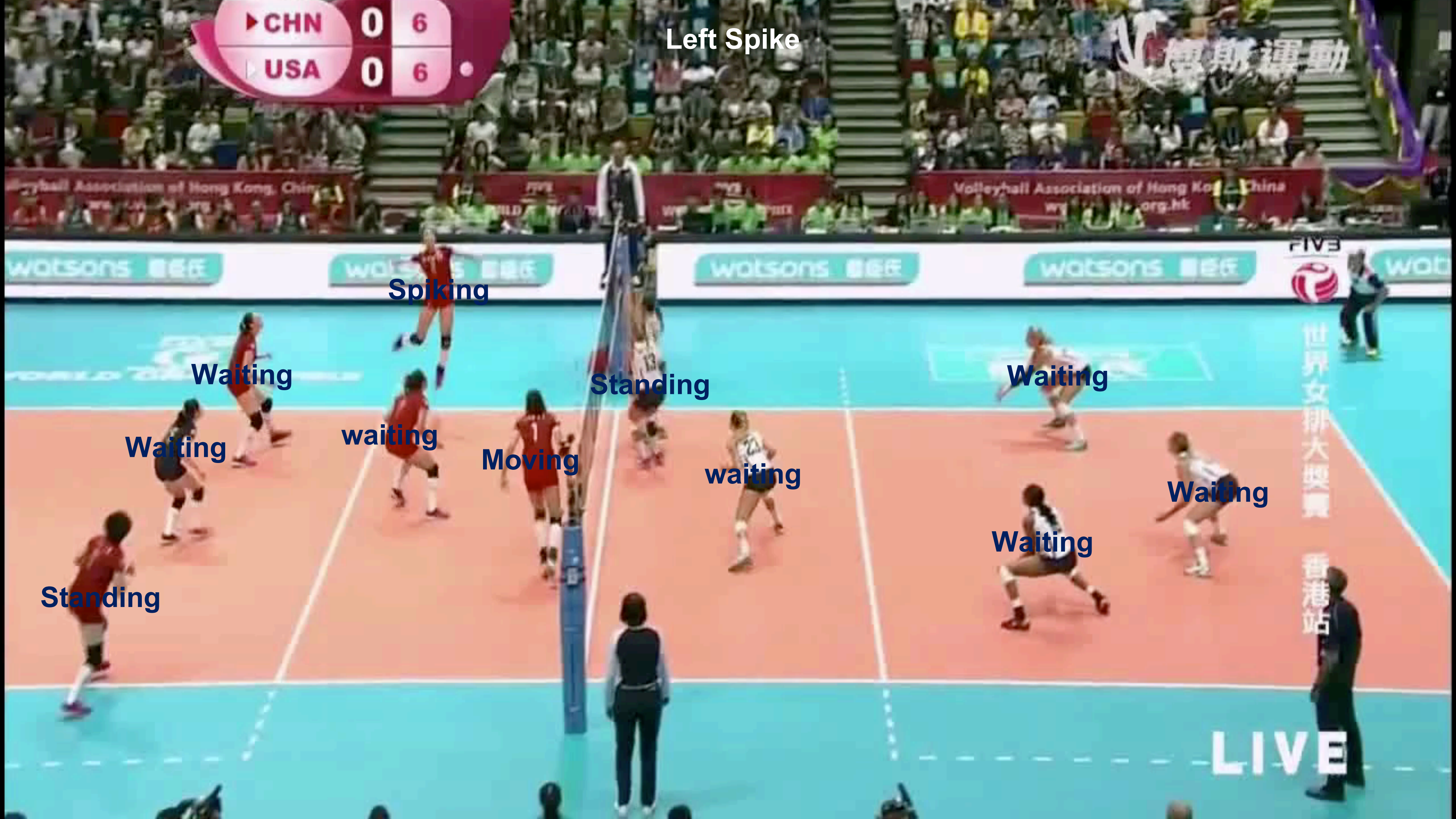

CHN 0 6
USA 0 6
Left Spike
Spiking
Waiting
Standing
Waiting
Waiting
waiting
Moving
waiting
Waiting
Waiting
Standing
LIVE

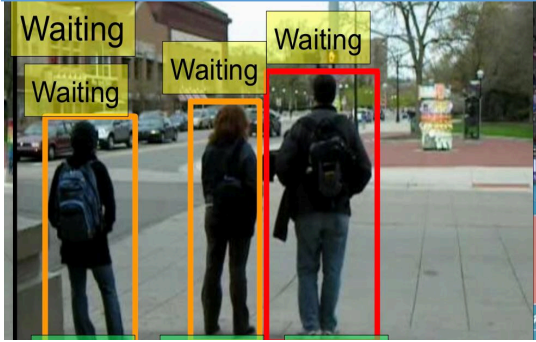
Image Classifier

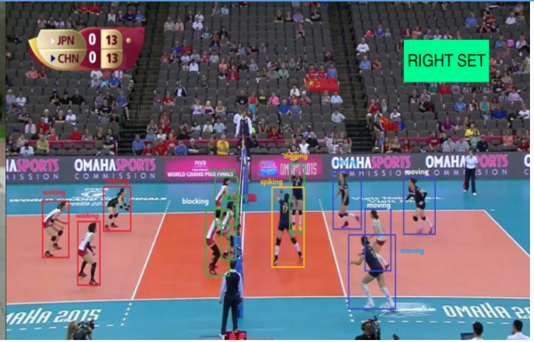
Group activity label

Challenge:

- high level description
- aggregate information over whole scene
- focus on relevant people

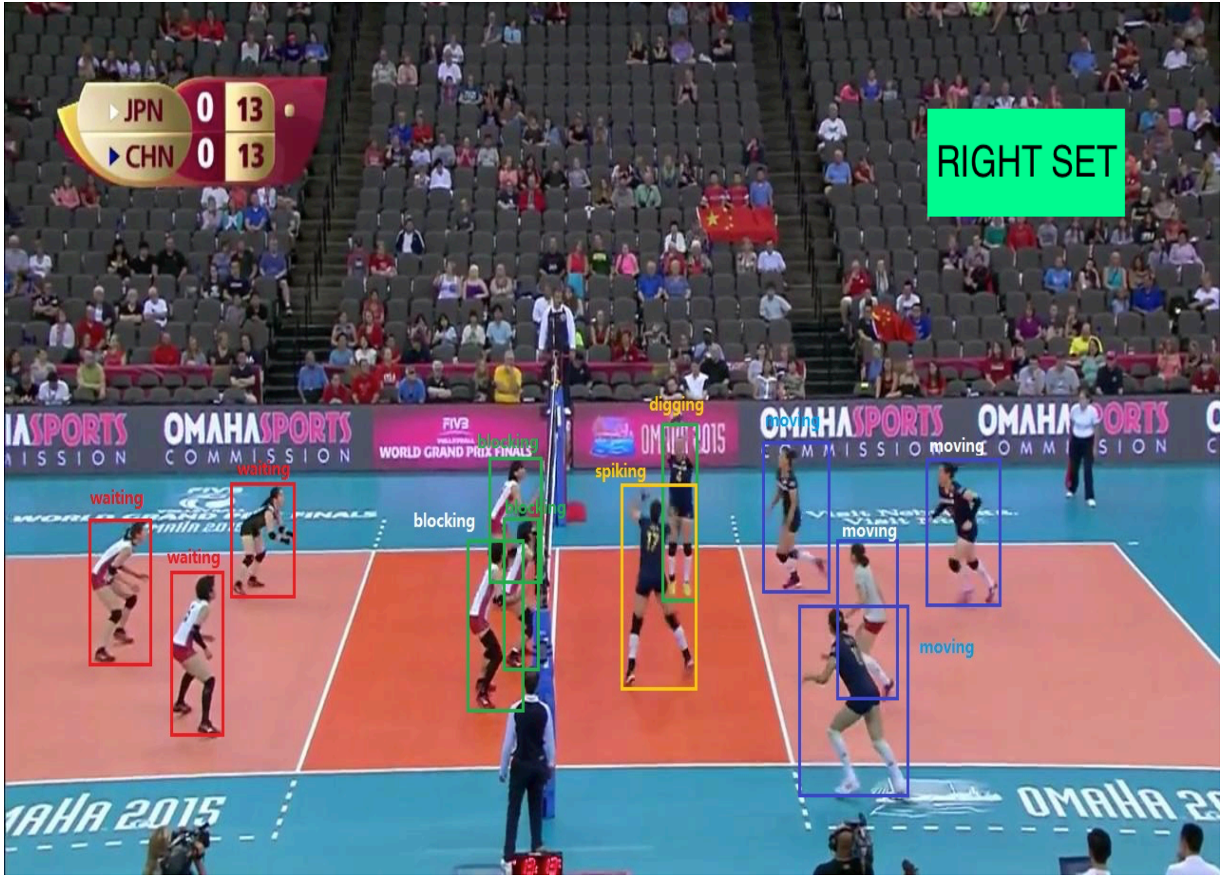
Group Activity = Majority's Activity	Group Activity = Key Player's Activity





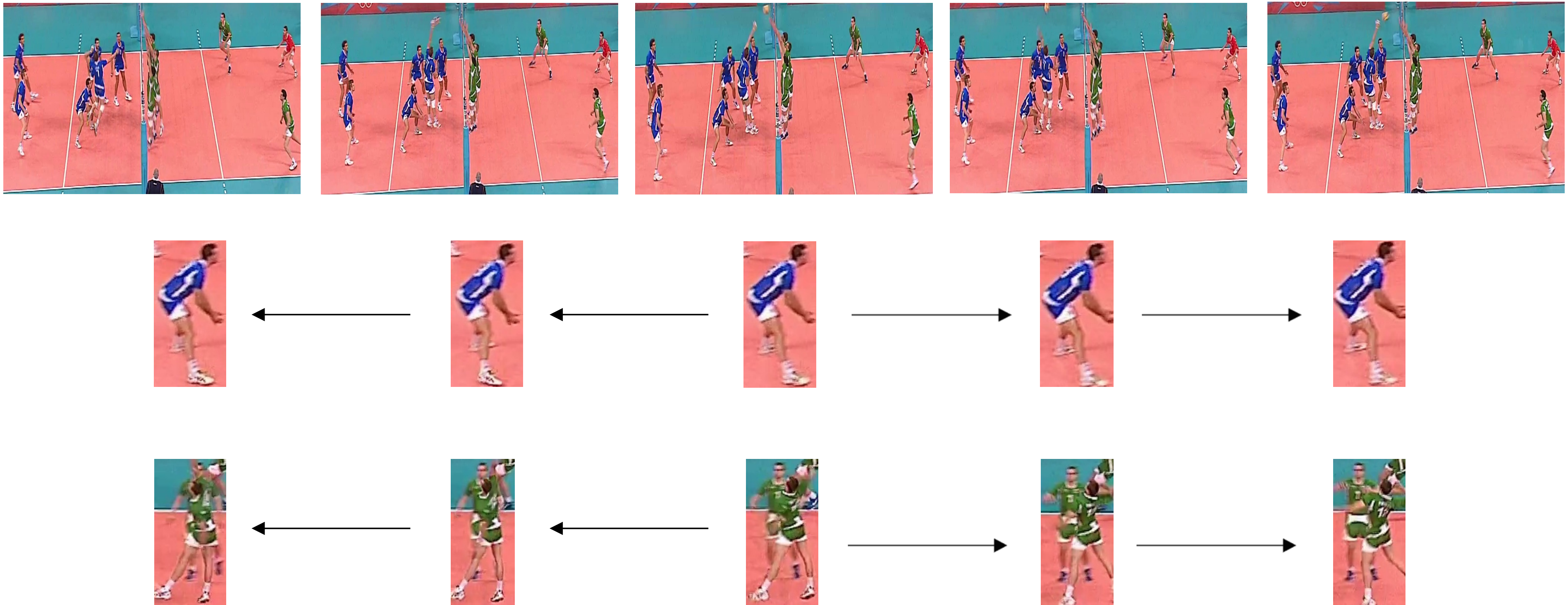
Group Activity – Right spike

Intuitive fix: use person-centric representation

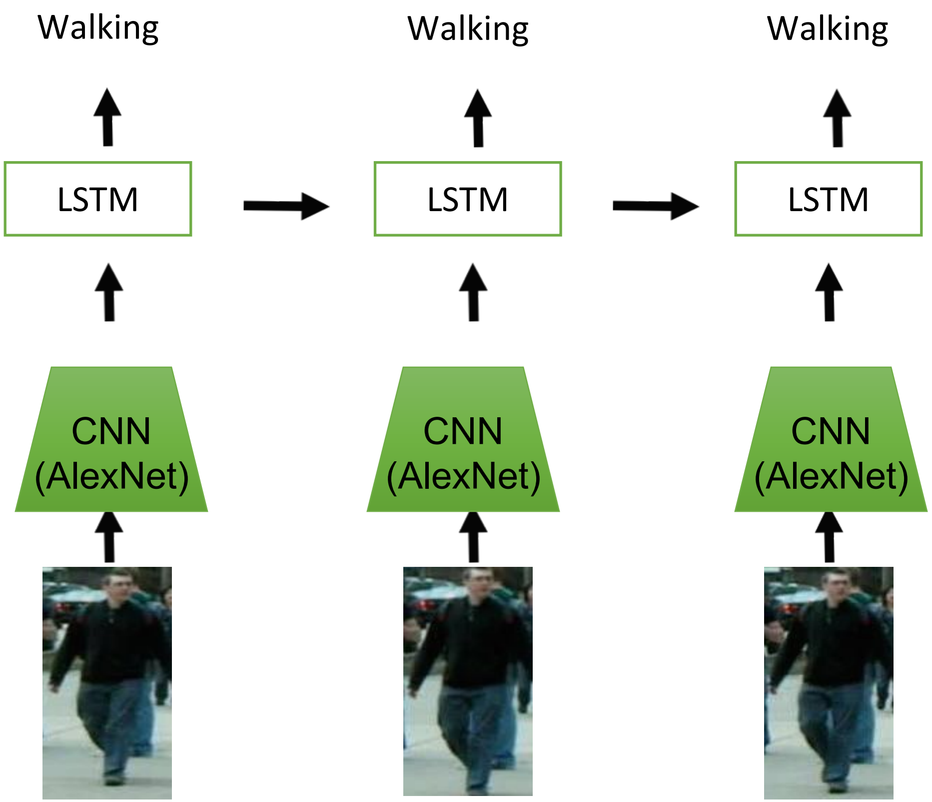


Person Tracks

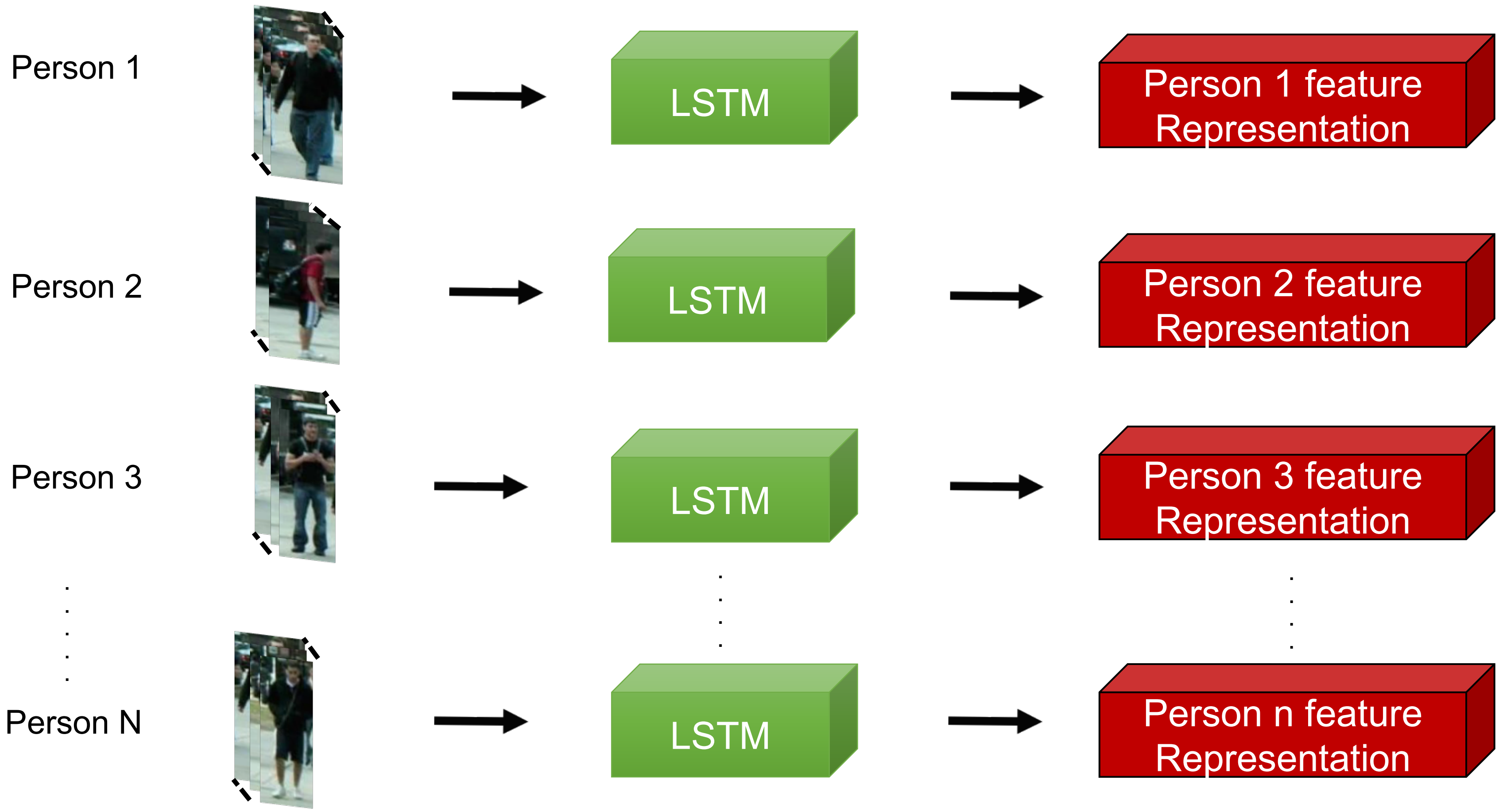
- Extract trajectories by tracking each person forward/backward in time



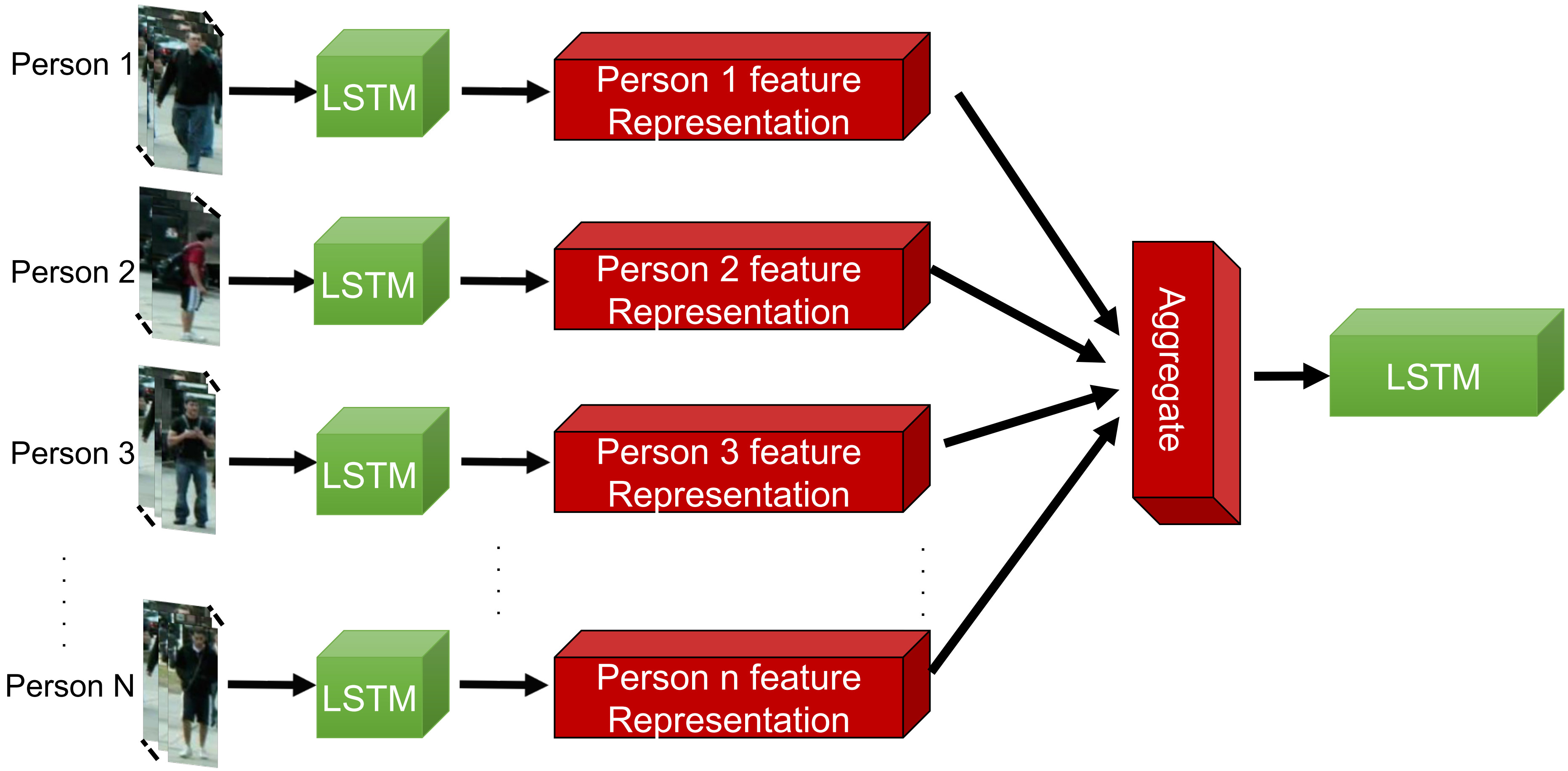
Stage 1 : Learning Individual Action Features



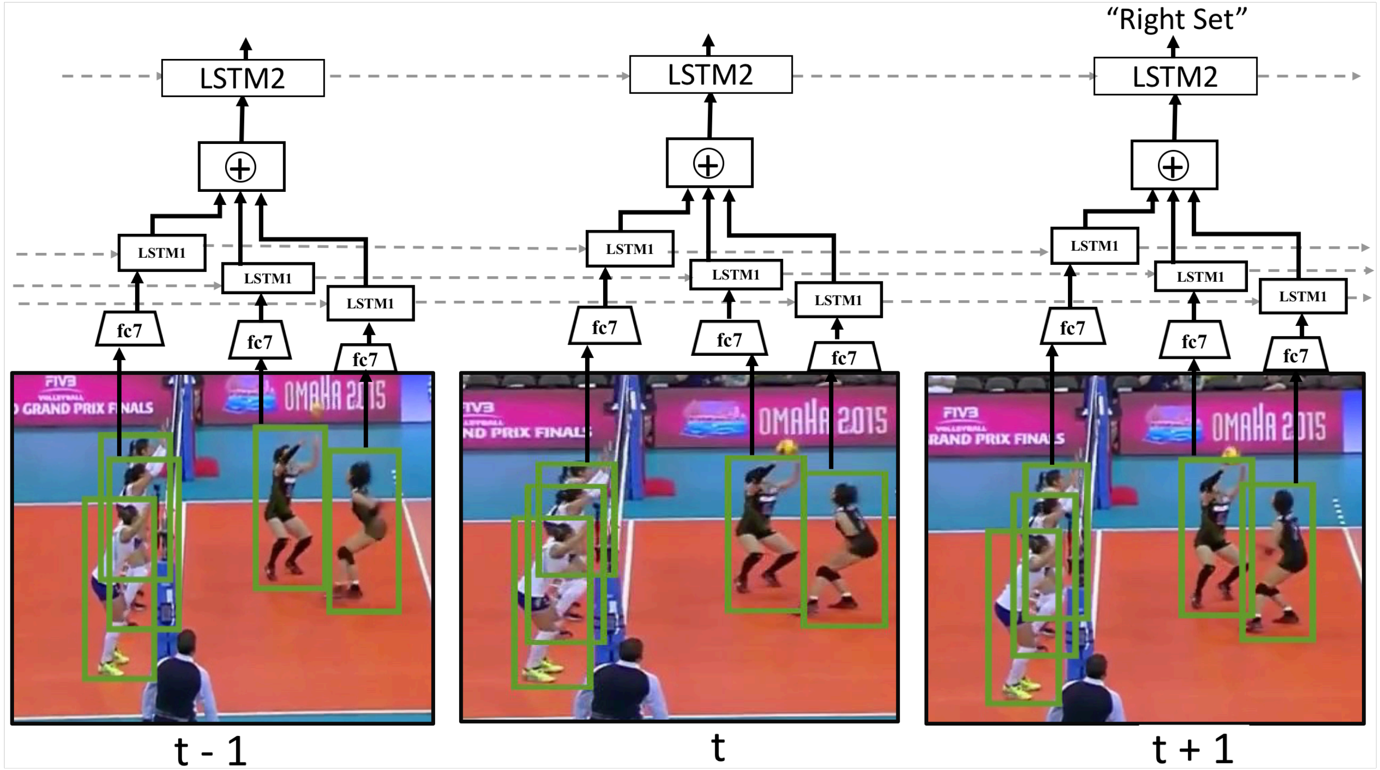
Stage1 : Learning Individual Action Features



Stage 2: Learning Frame Representations



Summary



"Right Set"
LSTM2
LSTM2
LSTM2
LSTM1
LSTM1
LSTM1
fc7
fc7
fc7
t - 1
t
t + 1

Collective Activity Dataset

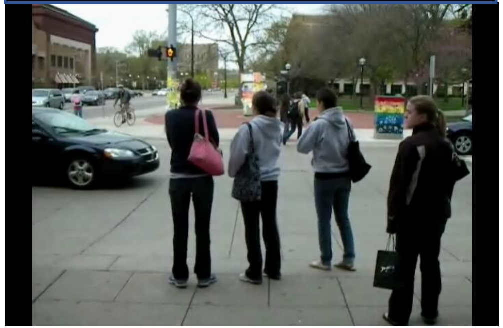
- Same label set for people and group activities
- 1925 video clips for training, 638 video clips for testing

1. Crossing

2. Queueing

3. Talking

4. Waiting



5. Walking

Choi et al., VSWS 2009

Collective Activity Dataset

Method	Accuracy
Image Classification	63.0
Person Classification	61.8
Person - Fine tuned	66.3
Temp Model - Person	62.2
Temp Model - Image	64.2
Our Model w/o LSTM1	70.1
Our Model w/o LSTM2	76.8
Our Model	81.5

Collective Activity Dataset

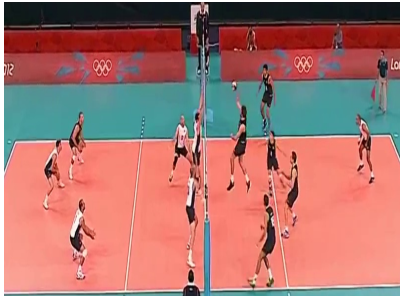
Method	Accuracy
Contextual Model [Lan et al. NIPS'10]	79.1
Deep Structured Model [Deng et al. BMVC'15]	80.6
Our Model	81.5
Cardinality Kernel [Hajimirsadeghi & Mori CVPR'15]	**83.4**

Method	Accuracy
Image Classification	63.0
Person Classification	61.8
Person - Fine tuned	66.3
Temp Model - Person	62.2
Temp Model - Image	64.2
Our Model w/o LSTM1	70.1
Our Model w/o LSTM2	76.8
Our Model	81.5

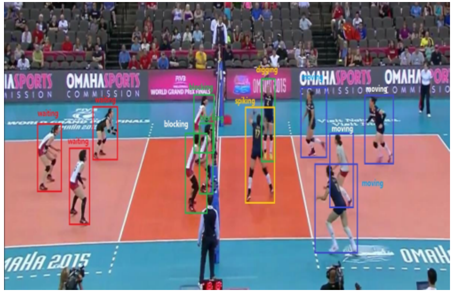
Volleyball Dataset – Frame Labels

- 4830 frames annotated from 55 volleyball videos
- 2/3 videos for training, 1/3 testing
- 9 player action labels
- 4 scene labels

Spiking

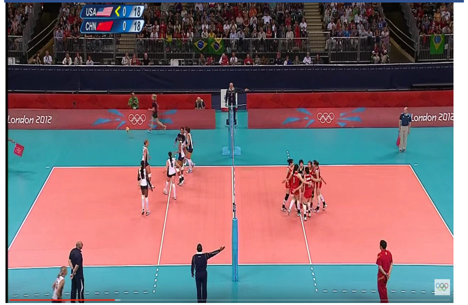


Setting



Passing

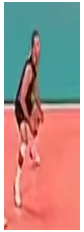
Win point



Left/right team variants

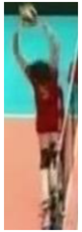
Volleyball Dataset – People Labels

Waiting

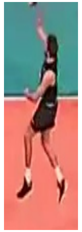


Digging

Setting



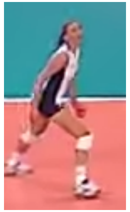
Spiking



Falling

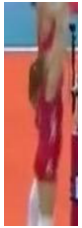
Jumping

Moving



Standing

Blocking



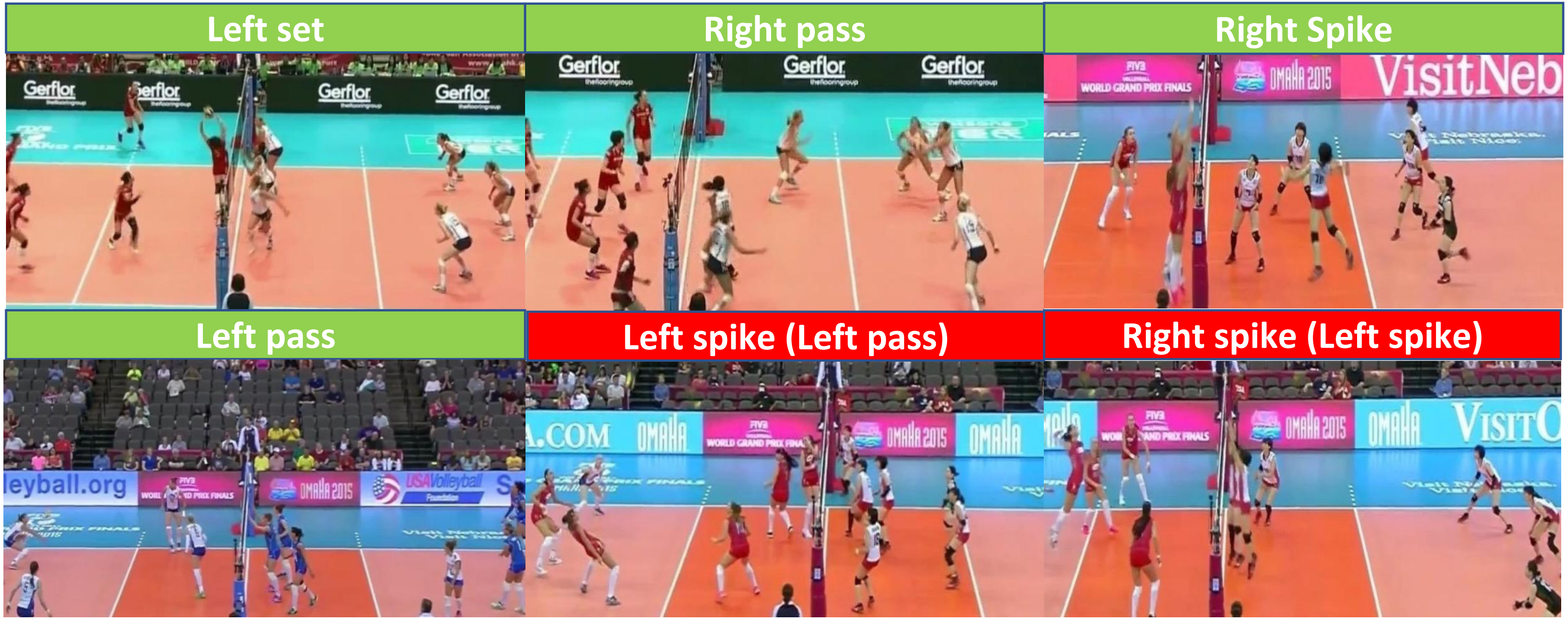
Experimental results on Volleyball Dataset

Method	Accuracy
Image Classification	66.7
Person Classification	64.5
Person - Fine tuned	66.8
Temp Model - Person	67.5
Temp Model - Image	63.1
Our Model w/o LSTM1	73.3
Our Model w/o LSTM2	80.9
Our Model	81.6

	lpass	rpass	lset	rset	lspike	rspike	lwin	rwin
lpass	79.65	3.98	9.73	0.00	3.10	2.65	0.44	0.44
rpass	4.29	80.00	0.00	9.52	2.86	1.90	0.95	0.48
lset	8.33	1.79	85.12	0.60	2.38	1.19	0.60	0.00
rset	5.21	19.27	1.04	68.23	0.00	4.69	1.56	0.00
lspike	3.35	1.12	5.03	0.00	89.94	0.56	0.00	0.00
rspike	2.31	5.20	2.31	4.62	1.16	83.24	1.16	0.00
lwin	2.94	3.92	0.00	0.00	0.00	0.00	88.24	4.90
rwin	1.15	1.15	0.00	0.00	0.00	0.00	12.64	85.06

Dense trajectories: 73.4-78.7

Visualization of results



Summary

- A two stage hierarchical model for group activity recognition

- LSTMs as a highly effective temporal model and temporal feature source

- People-relation modeling with simple pooling