视频动作理解：识别、检测与跟踪

报告人：王利民

南京大学计算机科学与技术系

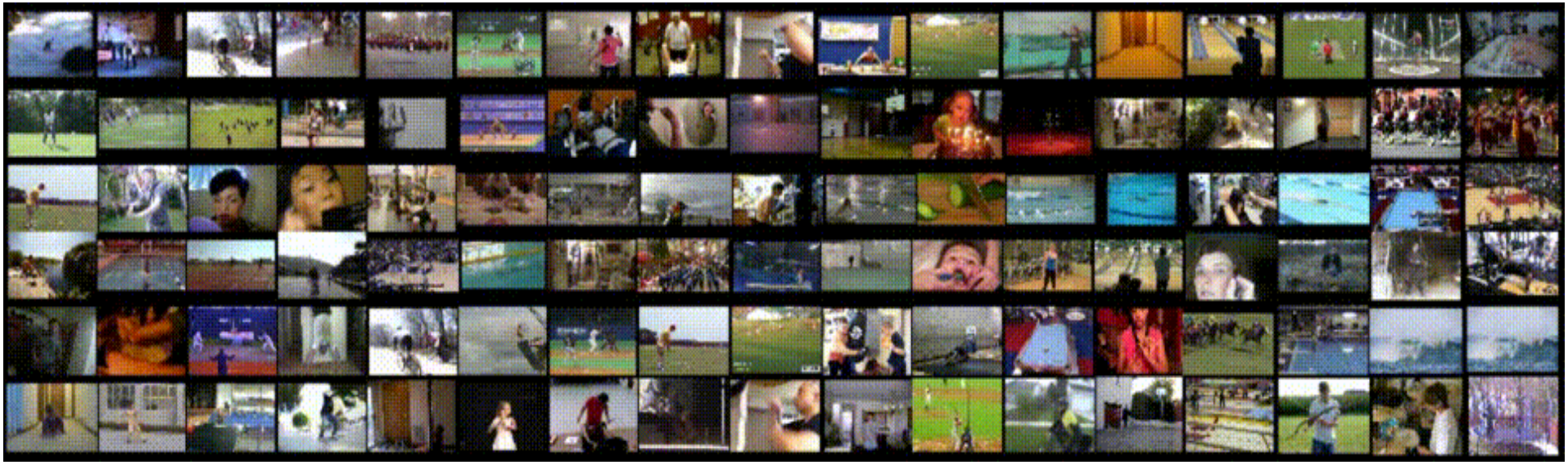
CCF-CV视界无限系列研讨会

北京工业大学 2020年11月8号

2020/11/10

MCG MULTIMEDIA COMPUTING GROUP 媒体计算研究组

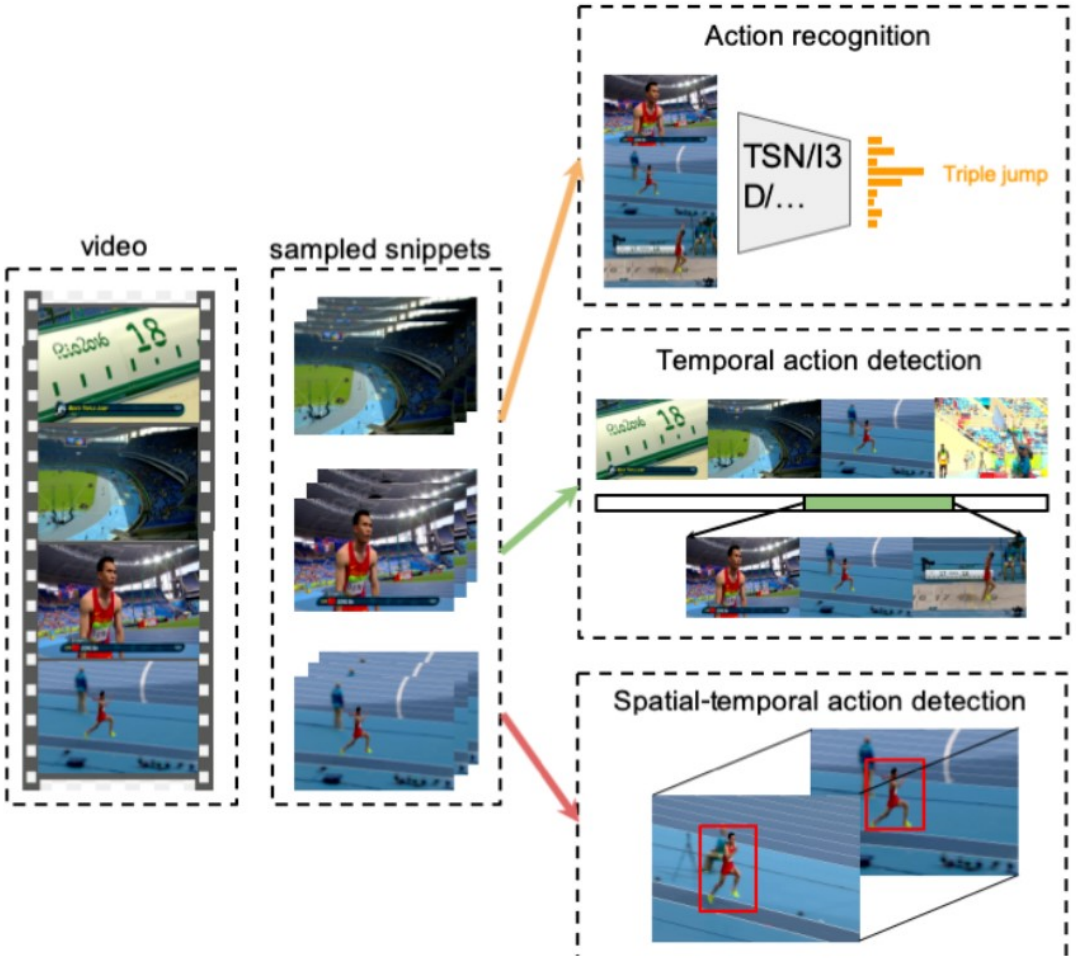
视频动作理解



- 学术界关注视频：实验室环境、电影/电视剧、互联网视频等等。
- 工业界关注视频：监控场景、工业流水线场景、直播场景等等。

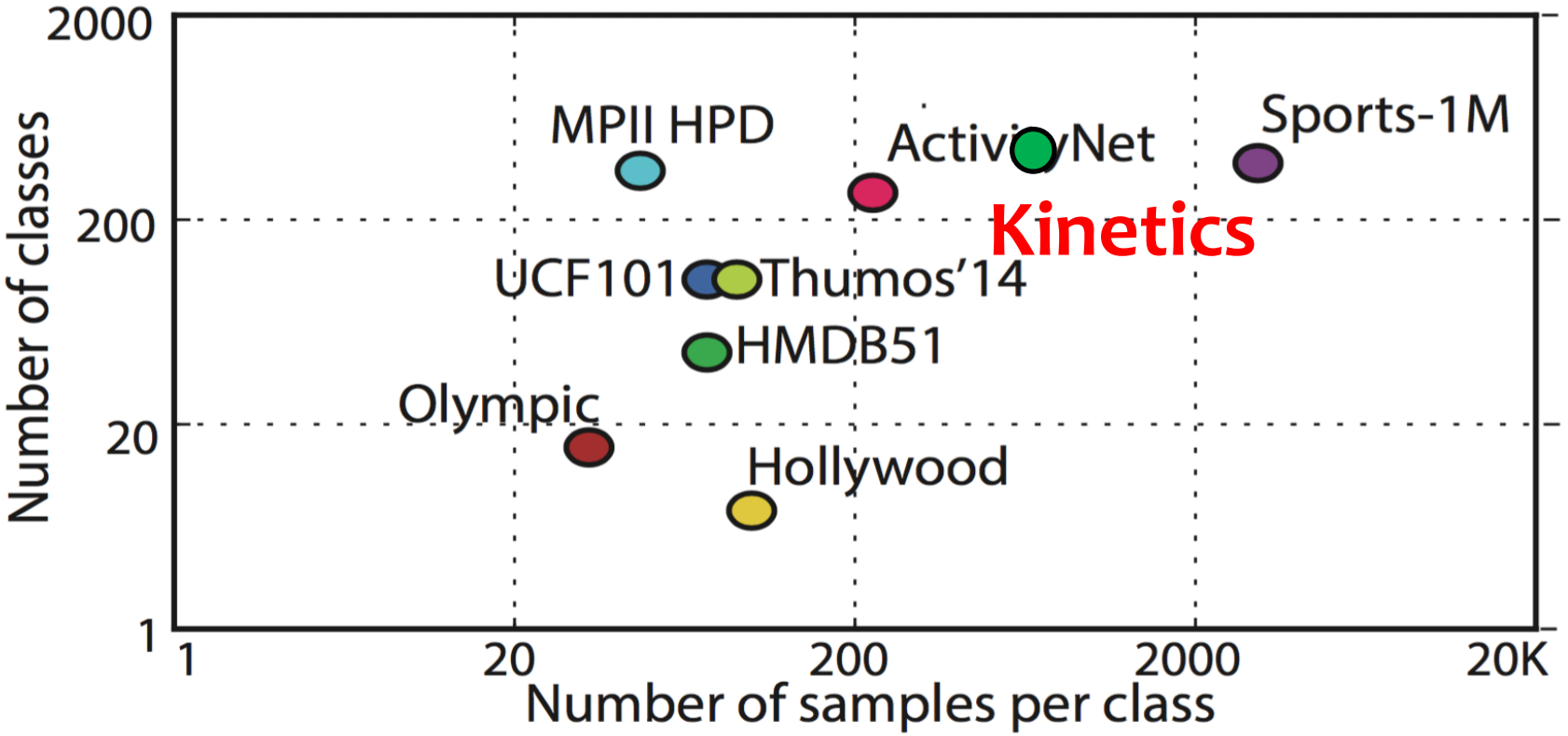
Haroon Idreeset al., *The THUMOS challenge on action recognition for videos "in the wild"*, in CVIU, 2017.

相关任务



- **Video Action Recognition**
  - Clip level Classification
- **Video Action Detection**
  - Temporal detection
  - Frame-level detection
  - Tube-level detection
- **Video Object Tracking**
  - Single object tracking
  - Multiple object tracking

学术界数据集

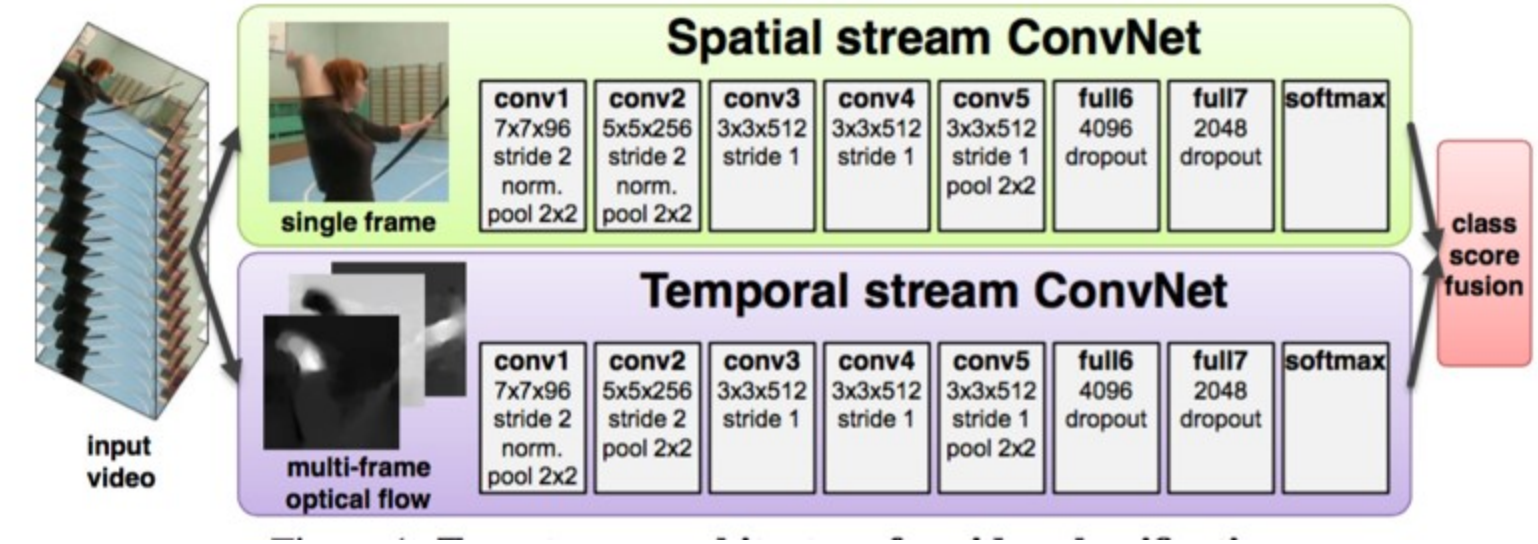

2000
200
20
1
Number of classes
MPII HPD
ActivityNet
Sports-1M
Kinetics
UCF101
Thumos'14
HMDB51
Olympic
Hollywood
1
20
200
2000
20K
Number of samples per class

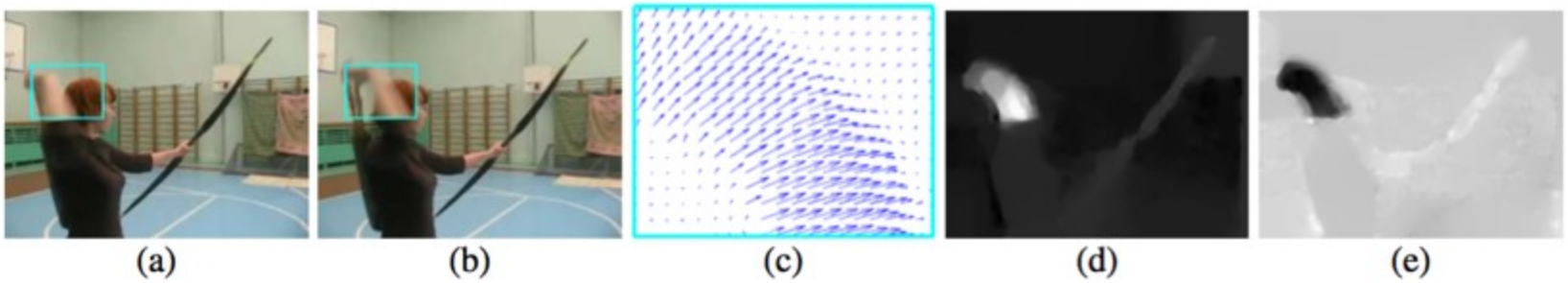
具体技术难点

MULTIMEDIA COMPUTING GROUP
媒体计算研究组

- **视频表征技术**
  - 时空特征的抽取（静态+动态）
  - 语义动作的分类（监督信号）
- **视频检测技术**
  - 时序定位：镜头切换，边缘歧义
  - 空间定位：不同尺度，遮挡，效率
- **视频跟踪技术**
  - 前后帧的关联（精度、平滑、相似物体）
  - 长时跟踪技术（遮挡、尺度，形变）

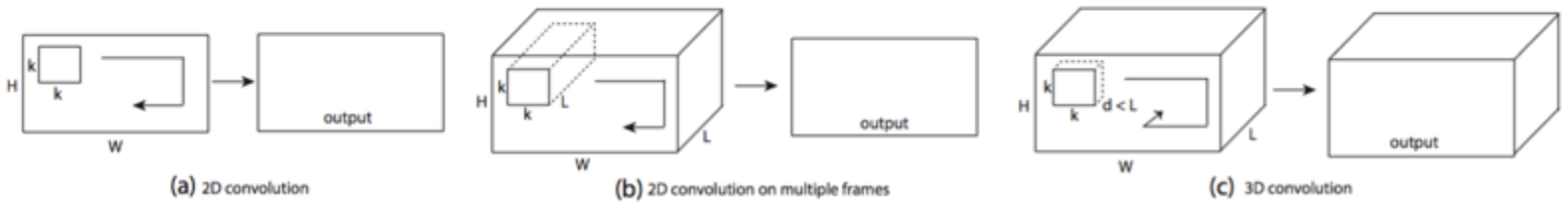
Two-stream CNN (2014)





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

3D CNN (2015)



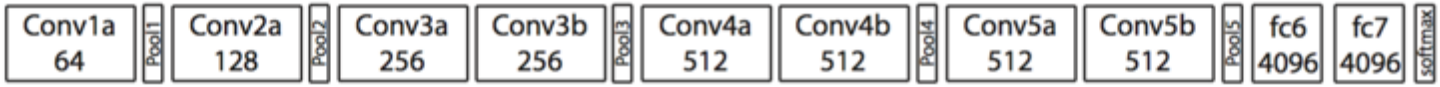


Figure 3. **C3D architecture**. C3D net has 8 convolution, 5 max-pooling, and 2 fully connected layers, followed by a softmax output layer. All 3D convolution kernels are $3 \times 3 \times 3$ with stride 1 in both spatial and temporal dimensions. Number of filters are denoted in each box. The 3D pooling layers are denoted from `pool1` to `pool5`. All pooling kernels are $2 \times 2 \times 2$, except for `pool1` is $1 \times 2 \times 2$. Each fully connected layer has 4096 output units.

Du Tran et al. *Learning Spatiotemporal Features with 3D Convolutional Networks*, in ICCV, 2015.

Appearance & Relation Net (2018)

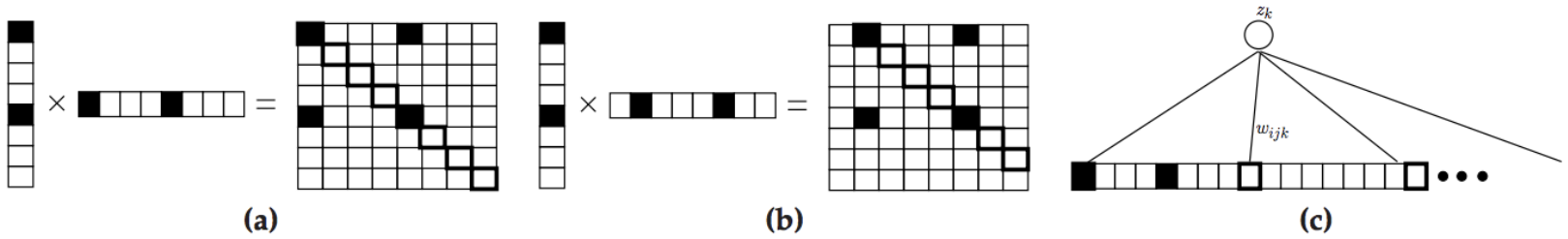
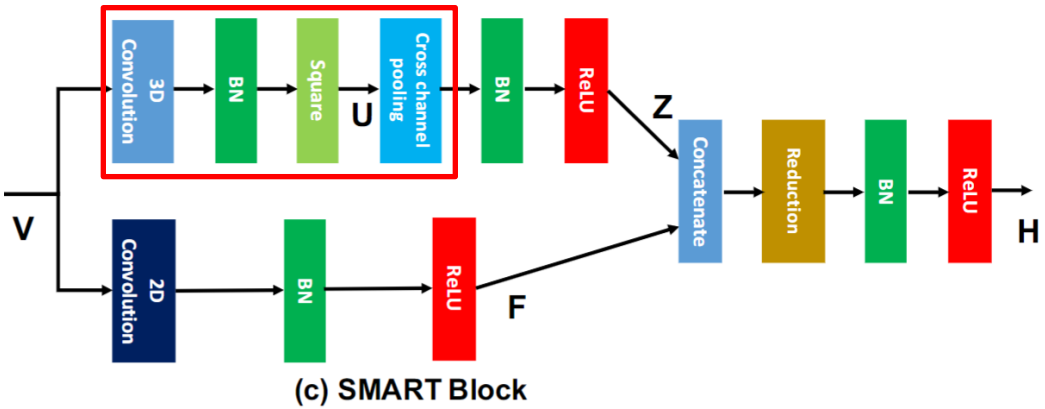


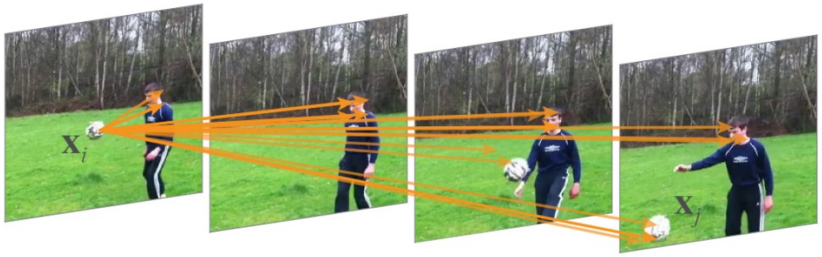
Fig. 3. **(a)** The diagonal of $L := xy^{\mathrm{T}}$ contains evidence for the identity transformation. **(b)** The secondary diagonals contain evidence for shifts. **(c)** A hidden unit that pools over one of the diagonals can detect transformations. Such a hidden unit would need to compute a *sum over products* of pixels.



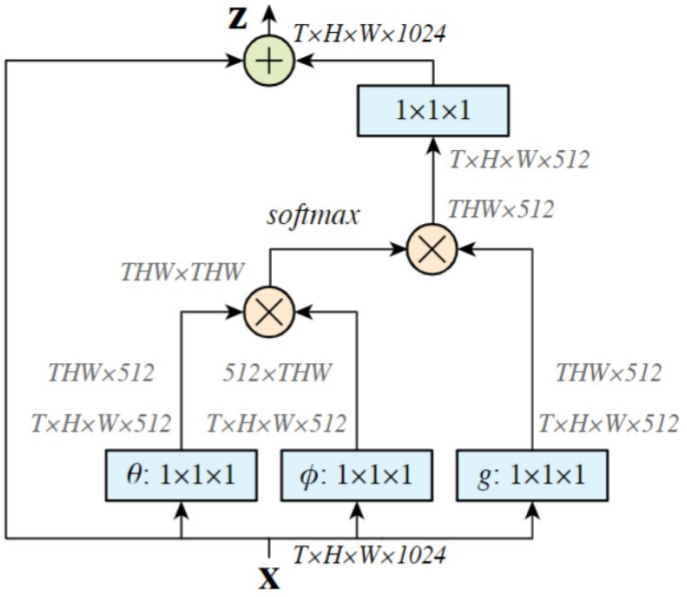
Method	Top-1	Top-5	Avg
C2D-ResNet18	61.2%	82.6%	71.9%
C3D-ResNet18	65.6%	85.7%	75.7%
C3D-ResNet34	67.1%	86.9%	77.0%
Relation-ResNet18 (s)	67.5%	87.0%	77.2%
Relation-ResNet18 (d)	67.1%	86.7%	76.9%
ARTNet-ResNet18 (s)	67.7%	87.1%	77.4%
ARTNet-ResNet18 (d)	**69.2%**	**88.3%**	**78.7%**

Limin Wang et al. *Appearance-and-Relation Networks for Video Classification*, in CVPR, 2018.

Non-local Net (2018)



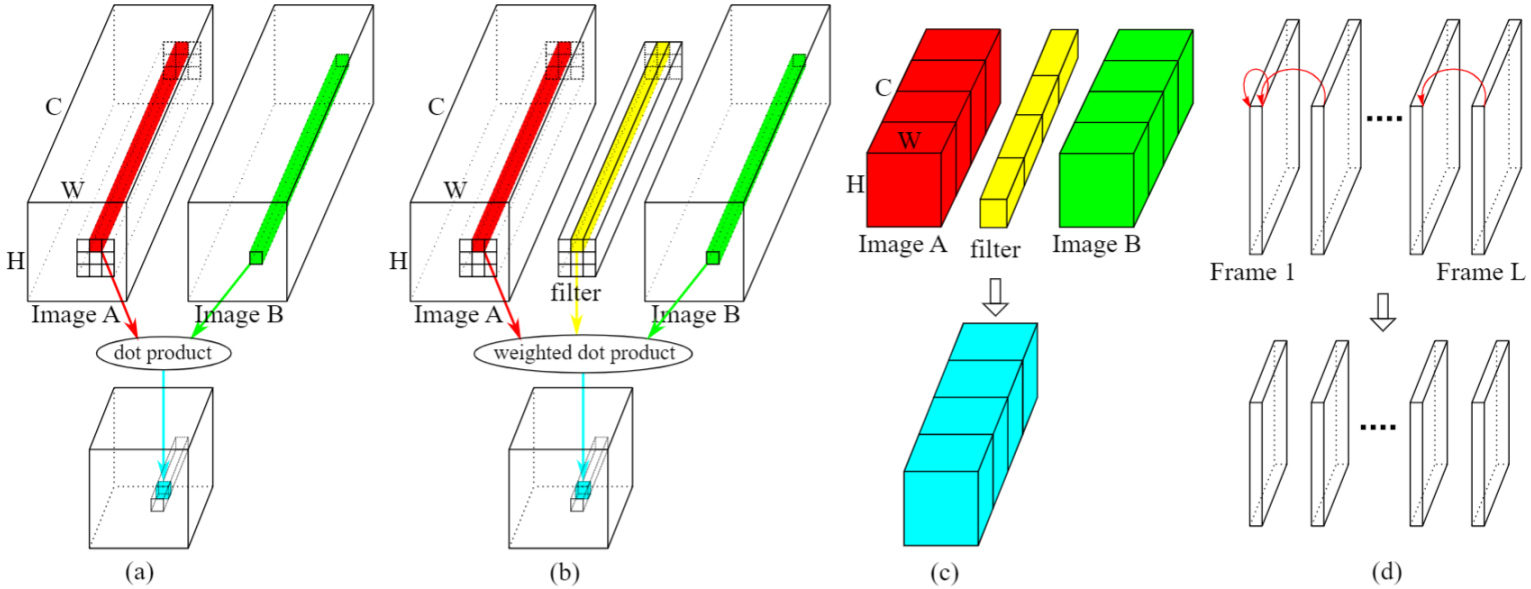
$$\mathbf{y}_i = \frac{1}{\mathcal{C}(\mathbf{x})} \sum_{\forall j} f(\mathbf{x}_i, \mathbf{x}_j) g(\mathbf{x}_j).$$

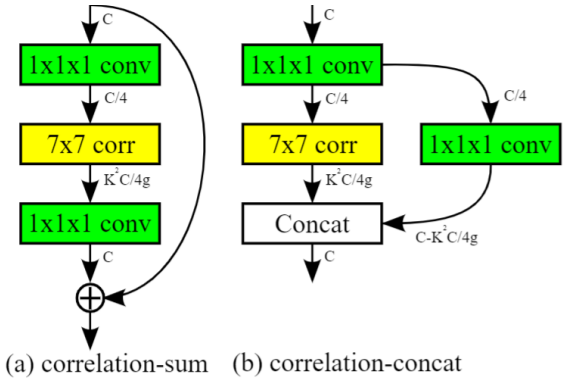


model, R101	params	FLOPs	top-1	top-5
C2D baseline	1×	1×	73.1	91.0
$\text{I3D}_{3\times3\times3}$	1.5×	1.8×	74.1	91.2
$\text{I3D}_{3\times1\times1}$	**1.2**×	1.5×	74.4	91.1
NL C2D, 5-block	**1.2**×	**1.2**×	**75.1**	**91.7**

Xiaolong Wang et al. *Non-local Neural Networks*, in CVPR, 2018.

Correlation Network (2020)





Model	Length	Top-1 accuracy (%) GFLOPs	Kinetics	Something	Diving
R2D-26	16	27.5	67.8	15.8	17.5
R(2+1)D-26	16	36.0	69.9	35.4	22.7
CorrNet-26	16	37.4	73.4	38.5	27.0
R2D-26	32	55.0	70.1	28.1	29.2
R(2+1)D-26	32	71.9	72.3	45.0	32.2
CorrNet-26	32	74.8	**75.1**	**47.4**	**35.5**

Heng Wang et al. Video Modeling with Correlation Networks, in CVPR, 2020.

Outline

- **视频表征模型**
  - TEINet: Towards an Efficient Architecture for Video Recognition (AAAI 2020)
  - TAM: Temporal Adaptive Module for Video Recognition (arXiv 2020)
- **视频检测框架**
  - Context-Aware RCNN：a Baseline for Action Detection in Videos (ECCV 2020)
  - Actions as Moving Points (ECCV 2020)
- **视频跟踪框架**
  - Fully convolutional online tracking (arXiv 2020)

Motivation

MULTIMEDIA COMPUTING GROUP
媒体计算研究组

- Temporal modeling is important for video recognition.
  - Two-stream CNNs
  - 3D CNNs

- 2D CNN + Lightweight Temporal Modeling
  - Self-attention modeling
  - Low computational cost
  - Pretrained on Image datasets
  - Benefit from the success of 2D CNNs

Enhance-and-Interact Scheme

- Motion information is able to identify discriminative moving object and people. (Channel)

- Temporal evolution of visual features enables us to capture dynamic semantics and relate adjacent features. (Temporal)

- **Enhance-and Interact scheme:** first enhance discriminative features and then capture their temporal interaction.

Pipeline

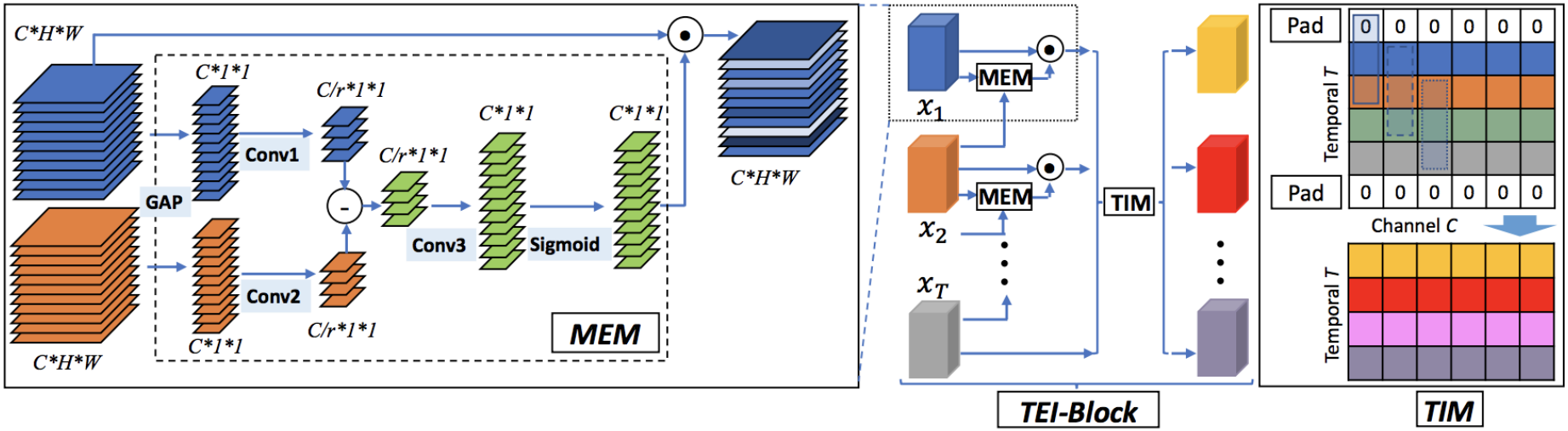


Figure 2: The pipeline of TEI block. We show motion enhanced module (**MEM**) in the left and temporal interaction module (**TIM**) in the right. The $\odot$ denotes element-wise multiplication, and $\ominus$ denotes element-wise subtraction. Notably, in TIM, we use different box to represent kernel weights, which means each channel do not share kernel weights.

MEM and TIM

- Motion Enhanced Module (MEM)

$$s_t = Conv1(\hat{x}_t, W_\theta) - Conv2(\hat{x}_{t+1}, W_\phi)$$

$$\hat{s}_t = \sigma(Conv3(s_t, W_\varphi))$$

$$u_t = \hat{s}_t \cdot x_t$$

- Temporal Interaction Module (TIM)

$$Y_{c,t,x,y} = \sum_i V_{c,i} \cdot \hat{U}_{c,t+i,x,y}$$

channel-wise temporal convolution

- TEI Block: MEM+TIM

Experiments

- TEINet for video recognition on several datasets:
  - Something-Something V1 & V2
  - Kinetics
- Backbone: 2D ResNet50
- Training: TSN Sampling & C3D Sampling (8 frame)
- Testing: 3 crops + 10 clips or center crop + 1 clip

Ablation Study

model	Top-1	Top-5
TSN	19.7%	46.6%
Res50+MEM	33.5%	61.5%
Res50+TIM	46.1%	74.7%
Res50+SE+TIM	46.1%	75.2%
Res50+MEM+TIM	**47.4%**	**76.6%**

(a) Exploring the impact of two modules and proving the importance of MEM.

stage	Top-1	Top-5
res_2	41.6%	70.1%
res_3	43.1%	72.1%
res_4	**45.4%**	**74.6%**
res_5	45.3%	74.3%

(b) How the TEI blocks in different stage of ResNet-50 influences the performance.

stages	Blocks	Top-1	Top-5
res_5	3	45.3%	74.3%
res_{4-5}	9	46.7%	76.3%
res_{3-5}	13	47.3%	75.2%
res_{2-5}	16	**47.4%**	**75.8%**

(c) The number of TEI block inserted into Network.

Table 5: Ablation studies on Something-Something V1.

Results on Kinetics

MULTIMEDIA COMPUTING GROUP
媒体计算研究组

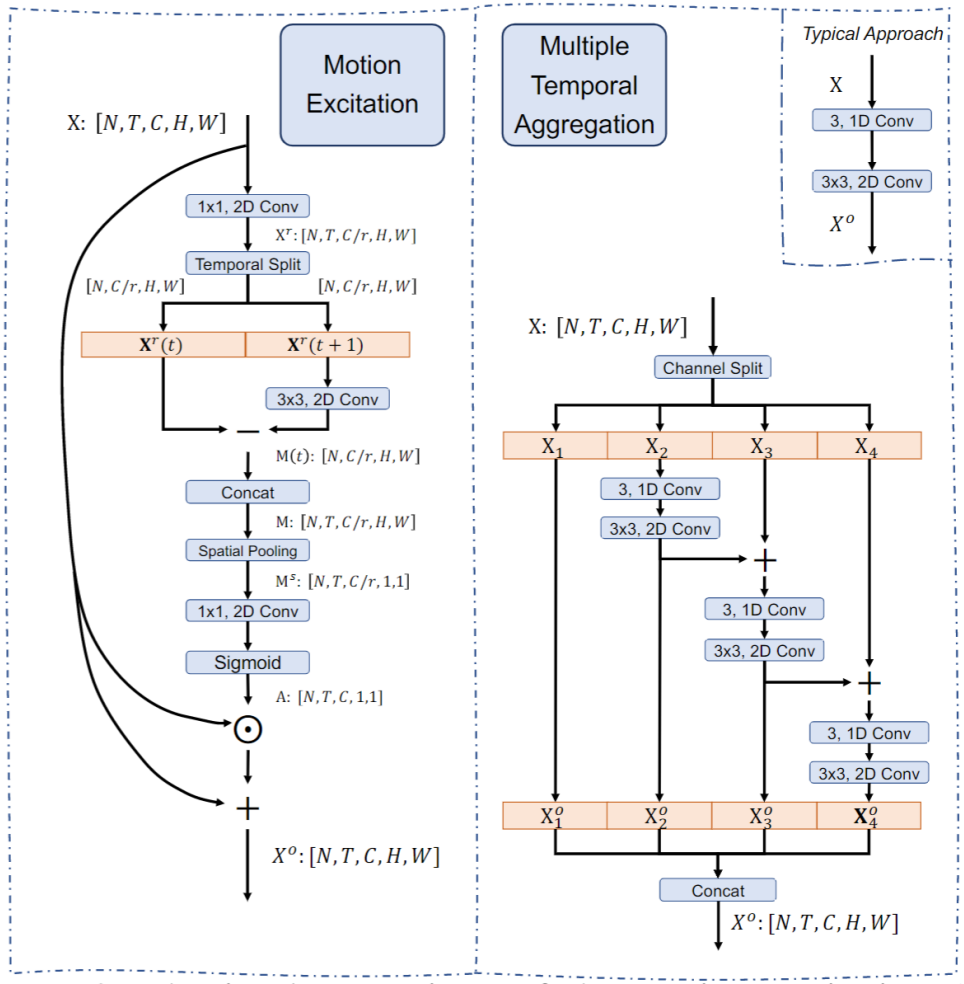
Method	Backbone	Pre-train	GFLOPs×views	Top-1	Top-5
$I3D_{64f}$ (Carreira et al. 2017)	Inception V1	ImgNet	108×N/A	72.1%	90.3%
$I3D_{64f}$ (Carreira et al. 2017)	Inception V1	From Scratch	108×N/A	67.5%	87.2%
NL+$I3D_{32f}$ (Wang et al. 2018b)	ResNet-50	ImgNet	70.5×30	74.9%	91.6%
NL+$I3D_{128f}$ (Wang et al. 2018b)	ResNet-50	ImgNet	282×30	76.5%	92.6%
NL+$I3D_{128f}$ (Wang et al. 2018b)	ResNet-101	ImgNet	359×30	77.7%	93.3%
$Slowfast_{8f}$ (Feichtenhofer et al. 2018)	ResNet-101	From Scratch	106×30	77.9%	93.2%
NL+$Slowfast_{16f}$ (Feichtenhofer et al. 2018)	ResNet-101	From Scratch	234×30	79.8%	93.9%
LGD-$3D_{128f}$ (Qiu et al. 2019)	ResNet-101	ImgNet	N/A×N/A	79.4%	94.4%
TSN (Wang et al. 2016)	BNInception	ImgNet	2.1×250	69.1%	88.7%
TSN (Wang et al. 2016)	Inception V3	ImgNet	3.2×250	72.5%	90.2%
ECO_{En} (Zolfaghari et al. 2018)	BNIncep+Res3D-18	From Scratch	N/A×N/A	70.7%	89.4%
$R(2+1)D_{32f}$ (Tran et al. 2018)	ResNet-34	Sports-1M	152×10	74.3%	91.4%
$R(2+1)D_{32f}$ (Tran et al. 2018)	ResNet-34	From Scratch	152×10	72.0%	90.0%
$ARTNet_{16f}$ (Wang et al. 2018a)	ResNet-18	From Scratch	23.5×250	69.2%	88.3%
S3D-G_{64f} (Xie et al. 2018)	Inception V1	ImgNet	71.4×30	74.7%	93.4%
$StNet_{25f}$ (He et al. 2019)	ResNet-101	ImgNet	310.5×1	71.4%	-
TSM_{16f} (Lin et al. 2018)	ResNet-50	ImgNet	65×30	74.7%	91.4%
$TEINet_{8f}$	ResNet-50	ImgNet	33×30	**74.9%**	**91.8%**
$TEINet_{16f}$	ResNet-50	ImgNet	66×30	**76.2%**	**92.5%**

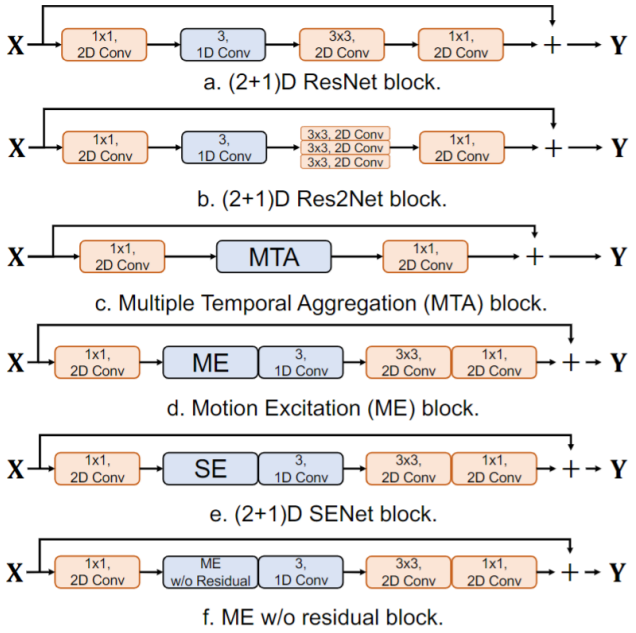
Results on Something-Something

MULTIMEDIA COMPUTING GROUP
媒体计算研究组

Method	Backbone	Pre-train	Frames	FLOPs	Val Top-1	Test Top-1
TSN-RGB (Wang et al. 2016)	BNInception	ImgNet	$8f$	16G	19.5%	-
TSN-RGB (Wang et al. 2016)	ResNet2D-50		$8f$	33G	19.7%	-
TRN-Multiscale-RGB (Zhou et al. 2018)	BNInception	ImgNet	$8f$	33G	34.4%	33.6%
TRN-Multiscale-RGB (Zhou et al. 2018)	ResNet2D-50		$8f$	33G	38.9%	-
TRN-Multiscale-2Stream (Zhou et al. 2018)	BNInception		$8f + 8f$	-	42.0%	40.7%
S3D-G-RGB (Xie et al. 2018)	Inception	ImgNet	$64f$	71G	48.2%	-
I3D-RGB (Wang and Gupta 2018)	ResNet3D-50	ImgNet+K400	$32f \times 2$	306G	41.6%	-
NL I3D-RGB (Wang and Gupta 2018)	ResNet3D-50			334G	44.4%	-
NL I3D+GCN-RGB (Wang and Gupta 2018)	ResNet3D-50+GCN			606G	46.1%	45.0%
ECO-RGB (Zolfaghari et al. 2018)	BNIncep+Res3D-18	K400	$16f$	64G	41.6%	-
ECO-RGB (Zolfaghari et al. 2018)			$92f$	267G	46.4%	-
ECO_{En}Lite-2Stream (Zolfaghari et al. 2018)			$92f + 92f$	-	49.5%	43.9
TSM-RGB (Lin, Gan, and Han 2018)	ResNet2D-50	ImgNet+K400	$8f$	33G	43.4%	-
TSM-RGB (Lin, Gan, and Han 2018)			$16f$	65G	44.8%	-
TSM_{En}-RGB (Lin, Gan, and Han 2018)			$16f + 8f$	98G	46.8%	-
TSM-2Stream (Lin, Gan, and Han 2018)			$16f + 16f$	-	50.2%	47.0
TEINet-RGB	ResNet2D-50	ImgNet	$8f$	33G	**47.4%**	-
			$8f \times 10$	330G	48.8%	-
			$16f$	66G	**49.9%**	-
			$16f \times 10$	660G	51.0%	47.5%
TEINet_{En}-RGB	ResNet2D-50	ImgNet	$16f + 8f$	99G	**52.5%**	48.1%

TEA: Extension of TEINet





Method	Frames×Crops×Clips	Val Top-1 (%)	Val Top-5 (%)
(2+1)D ResNet (a)[1]	8×1×1	46.0	75.3
(2+1)D Res2Net (b)[1]	8×1×1	46.2	75.5
MTA (c)[1]	8×1×1	47.5	76.4
TEA	8×1×1	**48.9**	**78.1**
(2+1)D ResNet (a)[1]	8×1×1	46.0	75.3
(2+1)D SENet (e)[1]	8×1×1	46.5	75.6
ME w/o Residual (f)[1]	8×1×1	47.2	76.1
STM [22][2]	8×1×1	47.5	-
ME (d)[1]	8×1×1	48.4	77.5
TEA	8×1×1	**48.9**	**78.1**

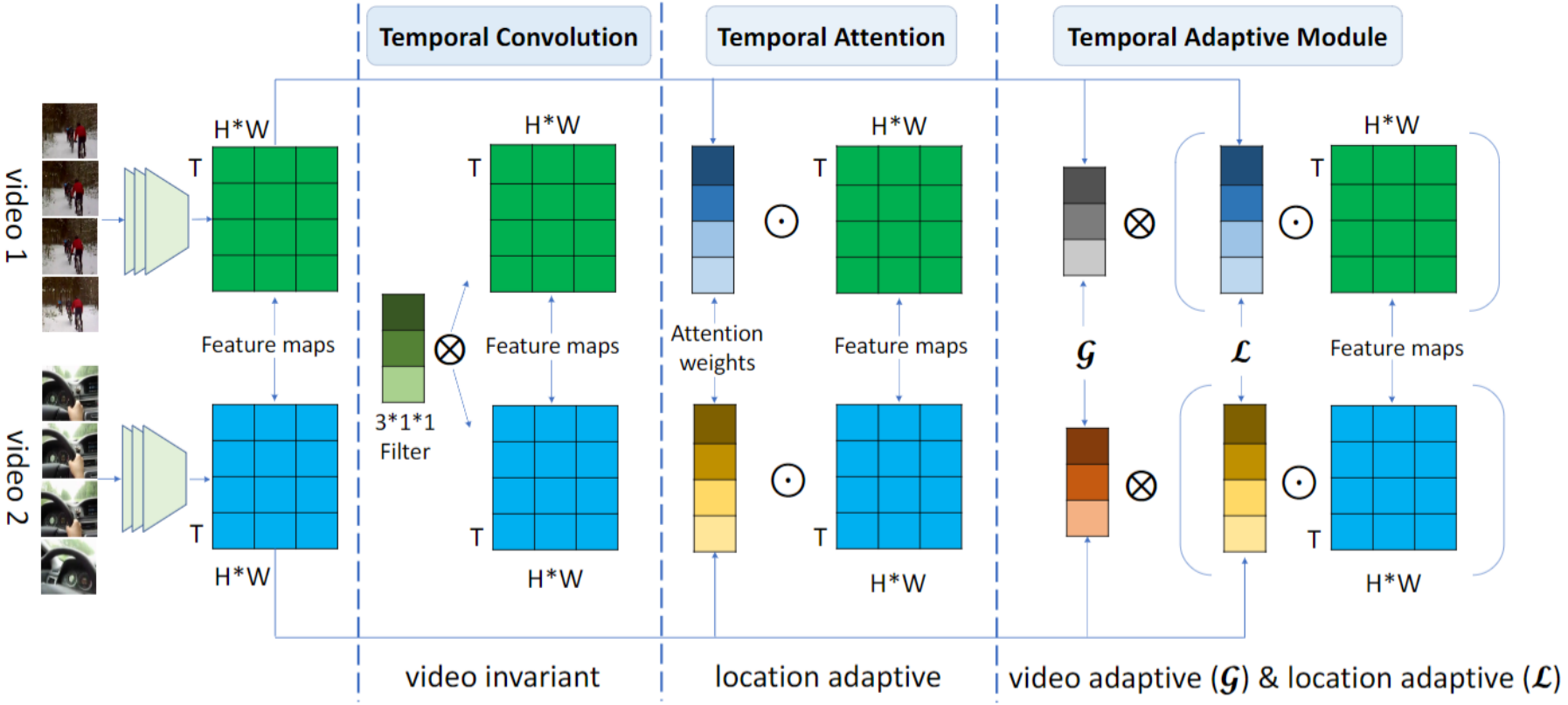
Y. Li et al., TEA: Temporal Excitation and Aggregation for Action Recognition, in CVPR 2020

Adaptive Temporal Modeling

MULTIMEDIA COMPUTING GROUP
媒体计算研究组

- Motion information is complex and diverse in videos.
- TEINet: MEM (Enhancement) + TIM (Interaction)
- Temporal adaptive modeling
  - Local adaptive modeling (location sensitive)
  - Global adaptive modeling (location invariant)

Overview



$$Y = \mathcal{G}(\hat{X}) \otimes (\mathcal{L}(\hat{X}) \odot X)$$

TAM: Local Branch

- Location sensitive weights:

$$V = \mathcal{L}(\hat{X}) = \text{Sigmoid}(\text{Conv1D}(\delta(\text{Conv1D}(\hat{X}, K, \frac{C}{\beta}), 1, C)))$$

- Weights replication:

$$\hat{V}_{c,t,j,i} = V_{c,t}$$

- Local enhancement:

$$Z = \hat{V} \odot X = \mathcal{L}(\hat{X}) \odot X$$

TAM: Global Branch

- Location invariant kernels:

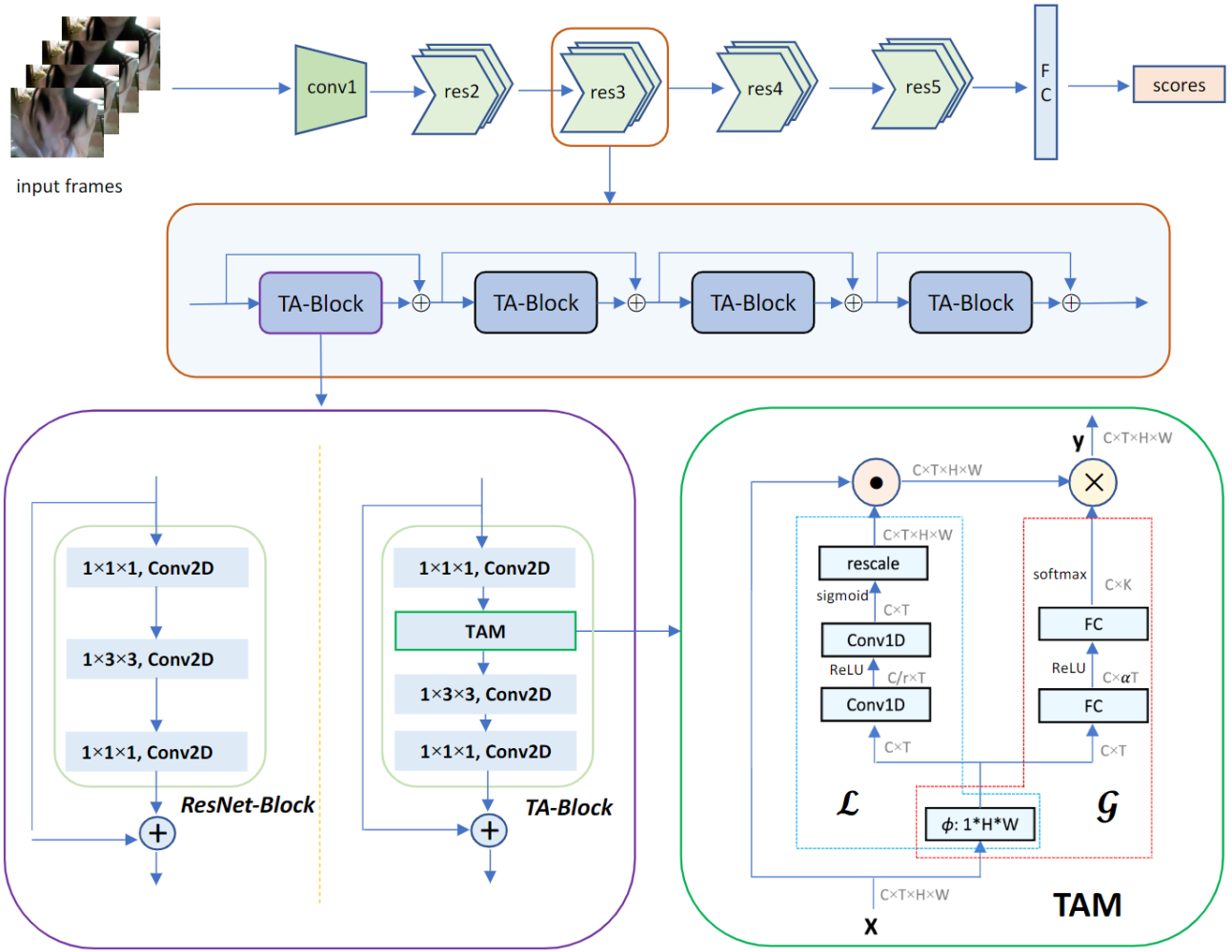
$$\Theta_c = \mathcal{G}(\hat{X})_c = \mathrm{softmax}(\mathcal{F}(\mathbf{W_2}, \delta(\mathcal{F}(\mathbf{W_1}, \hat{X}_c))))$$

- Temporal adaptive aggregation:

$$Y_{c,t,j,i} = \mathcal{G}(\hat{X}) \otimes Z = \Theta \otimes Z = \sum_k \Theta_{c,k} \cdot Z_{c,t+k,j,i}$$

- Channel-wise implementation

TANet


input frames
conv1
res2
res3
res4
res5
FC
scores
TA-Block
TA-Block
TA-Block
TA-Block
1×1×1, Conv2D
1×3×3, Conv2D
1×1×1, Conv2D
ResNet-Block
1×1×1, Conv2D
TAM
1×3×3, Conv2D
1×1×1, Conv2D
TA-Block
C×T×H×W
rescale
sigmoid
C×T
Conv1D
ReLU
C/r×T
Conv1D
C×T
ℒ
ϕ: 1*H*W
C×T×H×W
x
y
C×T×H×W
softmax
C×K
FC
ReLU
C×αT
FC
C×T
𝒢
TAM

Comparison of Temporal Module

Models	FLOPs (of single view)	Params	Top-1	Top-5
C2D	42.95G	24.33M	70.2%	88.9%
C2D-Pool	42.95G	24.33M	73.1%	90.6%
C2D-TConv	53.02G	28.10M	73.3%	90.7%
TSM (Lin et al., 2019)	42.95G	24.33M	74.1%	91.2%
TEINet (Liu et al., 2019b)	43.01G	25.11M	74.9%	91.8%
$I3D_{3\times1\times1}$ (Wang et al., 2018b)	62.55G	32.99M	74.3%	91.6%
NL C2D (Wang et al., 2018b)	64.49G	31.69M	74.4%	91.5%
Global branch	43.00G	24.33M	74.9%	91.7%
Local branch	43.00G	25.59M	73.3%	90.7%
Global branch + SE (Hu et al., 2018)	43.02G	24.65M	75.4%	92.0%
TANet	43.02G	25.59M	**76.1%**	**92.3%**

SOTA Comparison (Sth-Sth V2)

MULTIMEDIA COMPUTING GROUP
媒体计算研究组

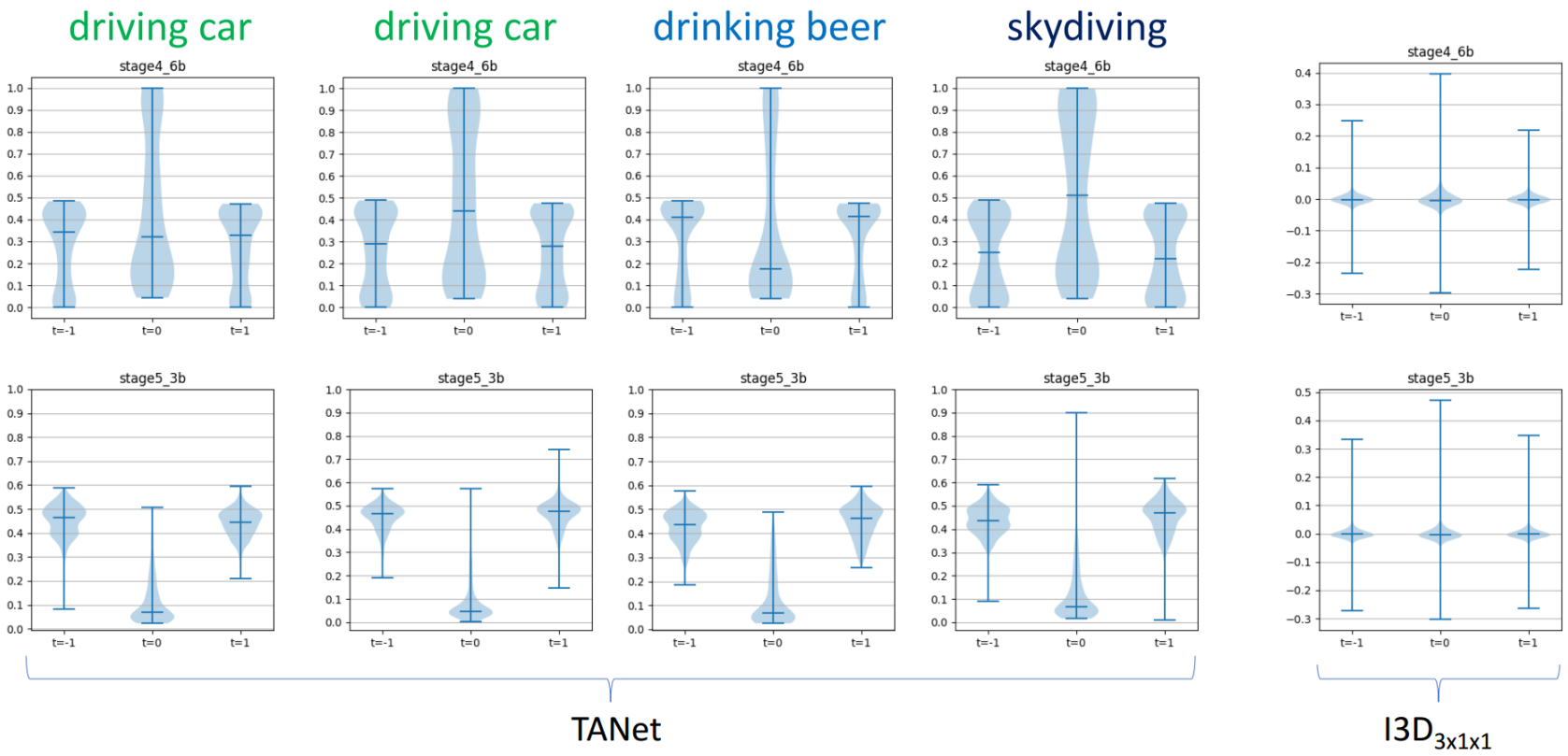
Methods	Backbones	Pre-train	frames×clips×crops	Top-1	Top-5
TRN (Zhou et al., 2018)	BNInception	ImgNet	$8f\times2\times3$	48.8%	77.6%
TSM (Lin et al., 2019)	ResNet50	ImgNet	$8f\times2\times3$	59.1%	85.6%
TSM (Lin et al., 2019)	ResNet50	ImgNet	$16\times2\times3$	63.4%	88.5%
$\text{TSM}_{RGB+Flow}$ (Lin et al., 2019)	ResNet50	ImgNet	$(16+16)\times2\times3$	66.0%	90.5%
CPNet (Liu et al., 2019a)	ResNet50	ImgNet	$24f\times16\times16$	57.7%	84.0%
GST (Luo & Yuille, 2019)	ResNet50	ImgNet	$8f\times1\times1$	61.6%	87.2%
GST (Luo & Yuille, 2019)	ResNet50	ImgNet	$16f\times1\times1$	62.6%	87.9%
bLVNet-TAM (Fan et al., 2019)	ResNet50	Sth-Sth V2	$32f\times1\times1$	61.7%	88.1%
TEINet Liu et al. (2019b)	ResNet50	ImgNet	$8f\times1\times1$	61.3%	-%
TEINet Liu et al. (2019b)	ResNet50	ImgNet	$16f\times1\times1$	62.1%	-%
TANet	ResNet50	ImgNet	$8f\times1\times1$	60.5%	86.2%
TANet	ResNet50	ImgNet	$8f\times2\times3$	62.7%	88.0%
TANet	ResNet50	ImgNet	$16\times1\times1$	62.5%	87.6%
TANet	ResNet50	ImgNet	$16\times2\times3$	64.6%	89.5%
TANet_{En}	ResNet50	ImgNet	$(8f+16f)\times2\times3$	**66.0%**	**90.1%**

SOTA Comparison (Kinetics 400)

MULTIMEDIA COMPUTING GROUP
媒体计算研究组

Methods	Backbones	Training Input	GFLOPs×views	Top-1	Top-5
TSN (Wang et al., 2016)	InceptionV3	3×224×224	3×250	72.5%	90.2%
ARTNet (Wang et al., 2018a)	ResNet18	16 ×112×112	23.5×250	70.7%	89.3%
S3D-G (Xie et al., 2018)	InceptionV1	64×224×224	71×30	74.7%	93.4%
I3D (Carreira & Zisserman, 2017)	InceptionV1	64×224×224	108×N/A	72.1%	90.3%
R(2+1)D (Tran et al., 2018)	ResNet34	32×112×112	152×10	74.3%	91.4%
NL I3D (Wang et al., 2018b)	ResNet50	32×224×224	N/A	74.9%	91.6%
NL I3D (Wang et al., 2018b)	ResNet50	128×224×224	282×30	76.5%	92.6%
TSM (Lin et al., 2019)	ResNet50	16×224×224	65×30	74.7%	91.4%
TEINet (Liu et al., 2019b)	ResNet50	16×224×224	86×30	76.2%	92.5%
bLVNet-TAM-24×2	bLResNet50	48×224×224	93×9	73.5%	91.2%
SlowOnly (Feichtenhofer et al., 2019)	ResNet50	8×224×224	42×30	74.8%	91.6%
SlowFast (Feichtenhofer et al., 2019)	ResNet50	(4+32)×224×224	36×30	75.6%	92.1%
TANet-50	ResNet50	8×224×224	43×30	76.1%	92.3%
TANet-50	ResNet50	16×224×224	86×12	**76.9%**	92.9%

TAM Visualization



Conclusion

- Motion information is complex and diverse in videos.

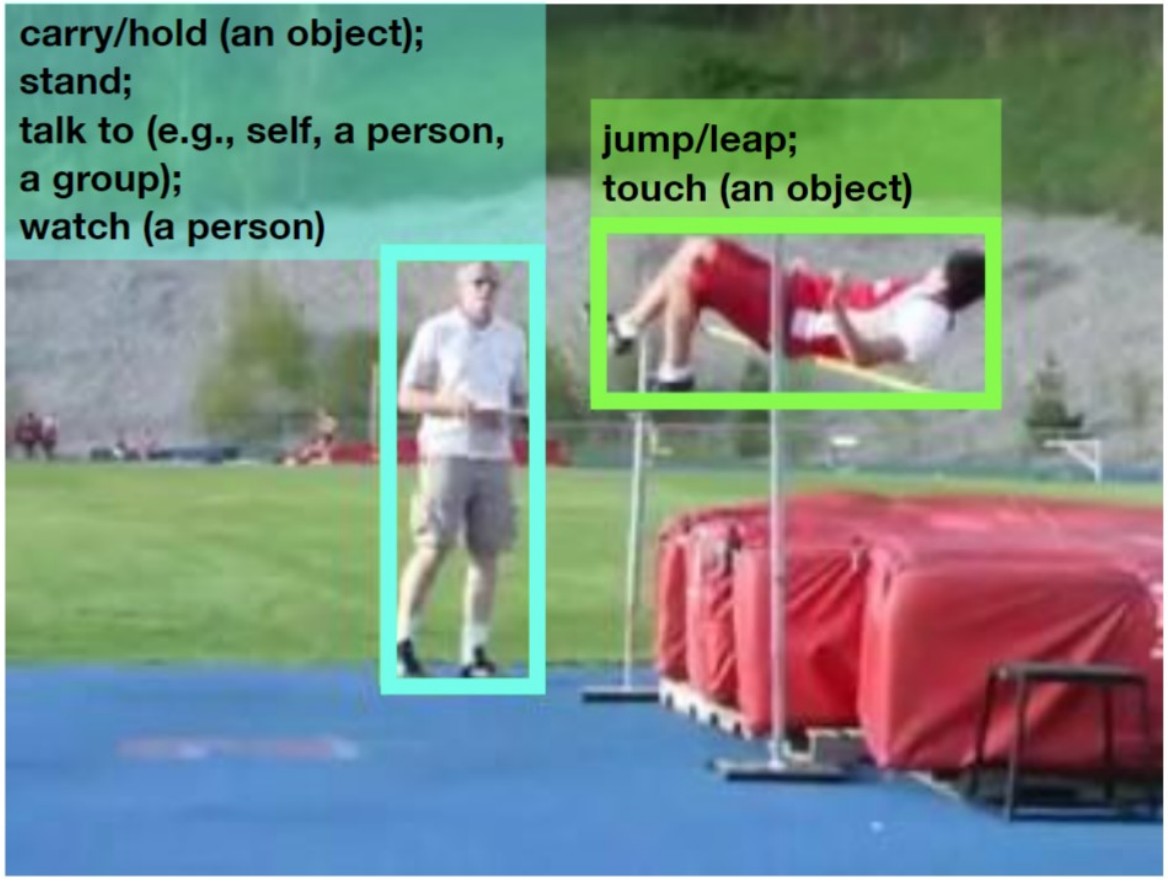
- Temporal adaptive modeling
  - Local adaptive modeling (location sensitive)
  - Global adaptive modeling (location invariant)
  - **Self-attention + Dynamic filtering**

- **Adaptive modeling** is useful to handle temporal complexity (Kinetics and Sth-Sth)

Outline

- **视频表征模型**
  - TEINet: Towards an Efficient Architecture for Video Recognition (AAAI 2020)
  - TAM: Temporal Adaptive Module for Video Recognition (arXiv 2020)
- **视频检测框架**
  - Context-Aware RCNN：a Baseline for Action Detection in Videos (ECCV 2020)
  - Actions as Moving Points (ECCV 2020)
- **视频跟踪框架**
  - Fully convolutional online tracking (arXiv 2020)

Frame-level Action Detection



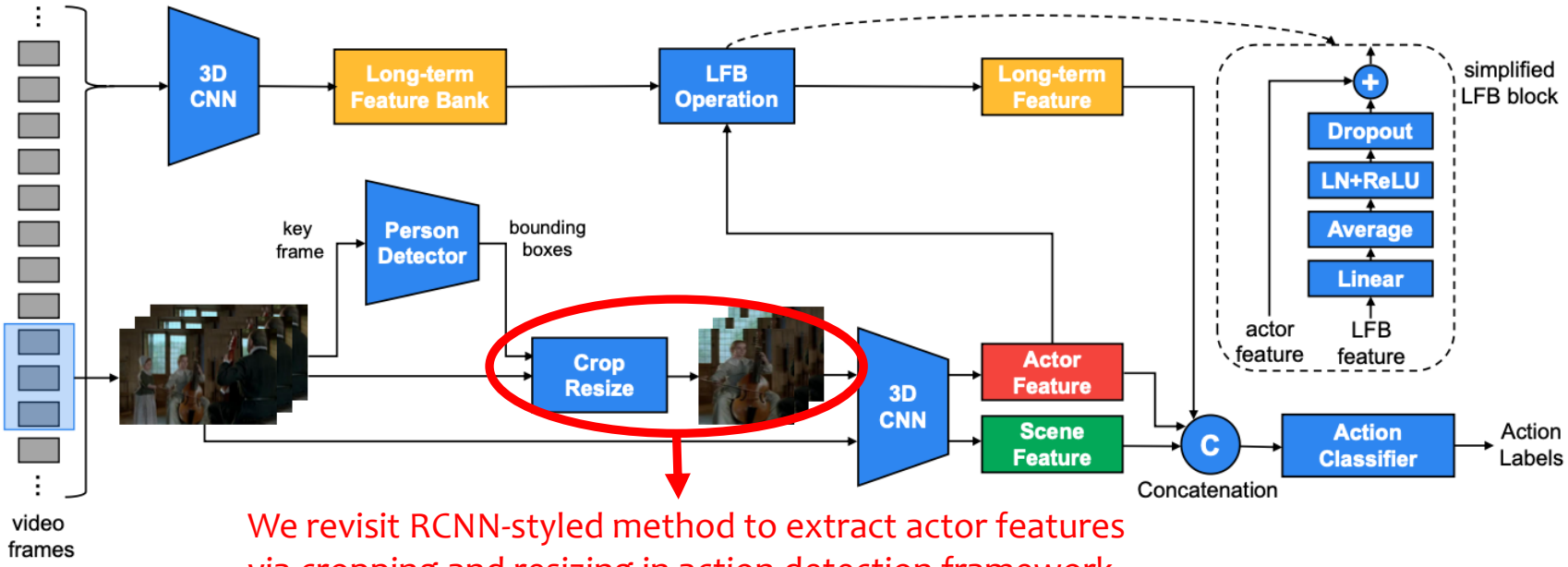
From Video Classification to Person-Level Classification

Action detection

- Spatio-temporal action detection
  - Localize actors and recognize their actions on untrimmed videos
  - Datasets: AVA, JHMDB, UCF-Sports
  - Evaluation Metrics: frame mAP and video mAP

Context-Aware RCNN

- Overall framework



We revisit RCNN-styled method to extract actor features via cropping and resizing in action detection framework

Technical details

- Actor features
  - Actor boxes are cropped directly from original video clip and resized to one fixed resolution as network input
- Scene features
  - Use a parameters shared network to extract the feature vector of the entire video clip
- Long-term features
  - All actor features centered at the current clip within window size of 60 seconds

RoI Pooling vs. Crop+Classification

- Performance comparison between RoI-Pooling and RCNN-based methods on the AVA dataset

Method	$T \times \tau$	mAP
RoI Pooling	8×8	20.1
	16×4	21.9
	32×2	22.1
Crop+Resize	8×8	23.1
	16×4	24.7
	32×2	25.0

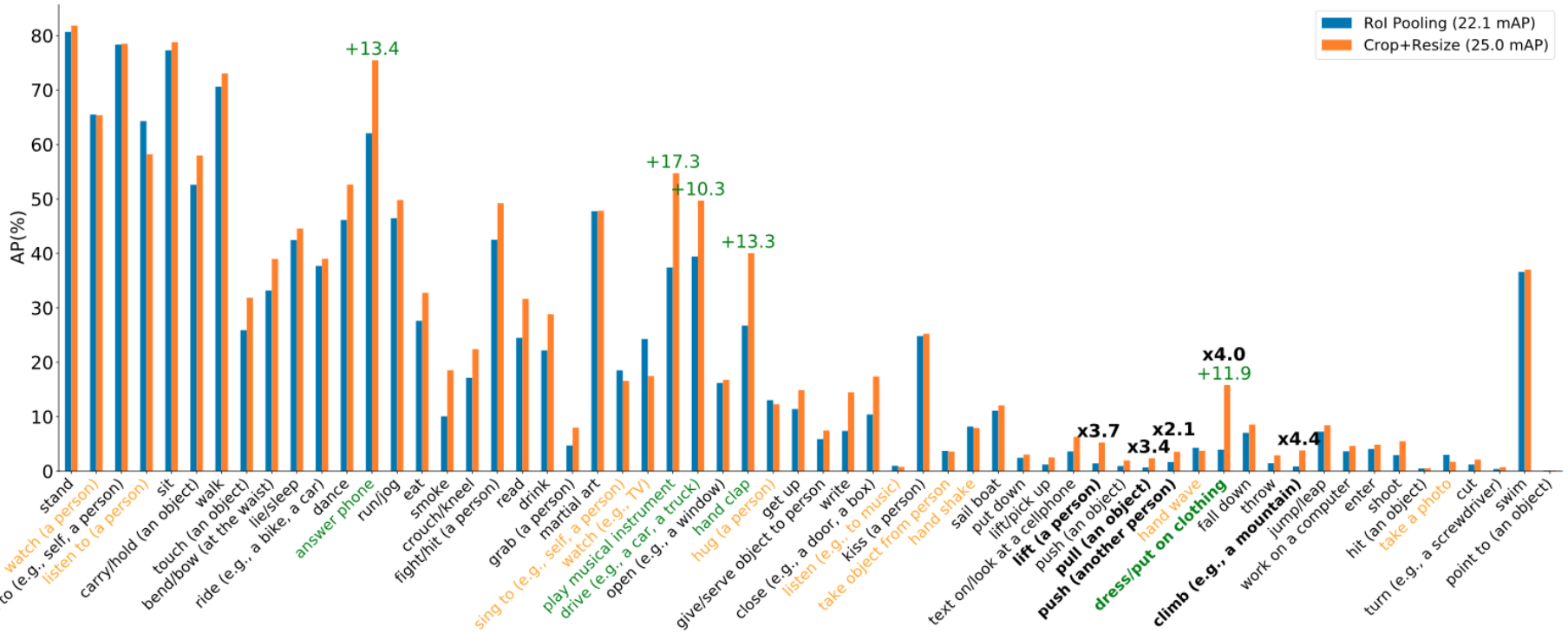
(a) Performance comparison with different input frames.

Method \ Size	Extra small	Small	Medium	Large	Extra large
RoI Pooling	12.9	16.2	18.3	21.1	22.2
Crop+Resize	16.9	19.2	20.1	21.8	22.0
Improvement	+4.0	+3.0	+1.8	+0.7	−0.2

(b) Performance comparison with different actor box sizes.

Experiment results

- Per-category performance comparison on the AVA dataset



Context feature fusion

- Performance using scene features and long-term features.

Scene feature	Long-term feature	mAP
-	-	24.7
Concat	-	25.7
Transformer+concat	-	25.8
Concat	NL attention	27.6
Concat	NL average	27.8
Concat	NL average w/o last linear	28.0

Experiment results

- Comparison with state-of-the-arts

Method	Flow	Video Pretrain	Backbone	mAP
AVA baseline [9]	✓	Kinetics-400	I3D	15.6
ACRN [29]	✓	Kinetics-400	S3D	17.4
Relation Graph [38]		Kinetics-400	R50-NL	22.2
VAT [6]		Kinetics-400	I3D	25.0
SlowFast [5]		Kinetics-400	R50	24.2
SlowFast [5]		Kinetics-400	R101	26.3
SlowFast [5]		Kinetics-600	R101-NL	28.2
LFB [36]		Kinetics-400	R50-NL	25.8
LFB [36]		Kinetics-400	R101-NL	27.1
ours		Kinetics-400	R50-NL	28.0

(a) Comparison with state-of-the-art on the AVA v2.1.

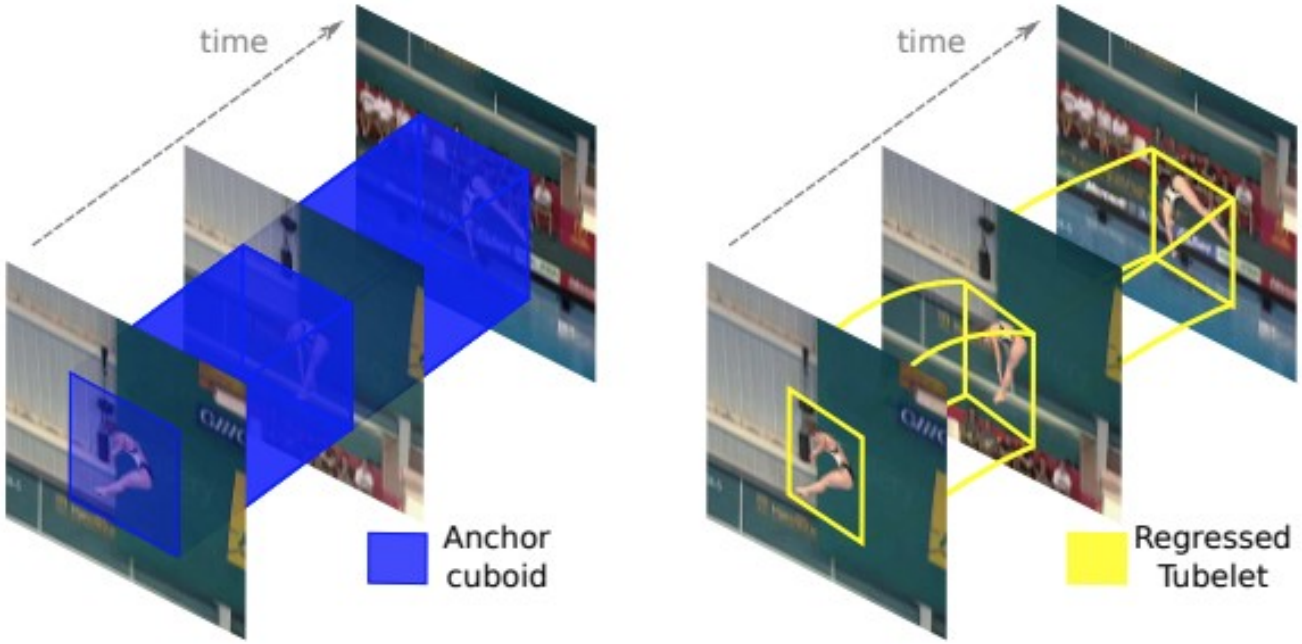
Method	Flow	Video Pretrain	Backbone	mAP
Two-stream RCNN [23]	✓		VGG	58.5
T-CNN [13]			C3D	61.3
ACT [16]	✓		VGG	65.7
AVA baseline [9]	✓	Kinetics-400	I3D	73.3
ACRN [29]	✓	Kinetics-400	S3D	77.9
ours		Kinetics-400	R50-NL	79.2

(b) Comparison with state-of-the-art on the JHMDB dataset.

Conclusion

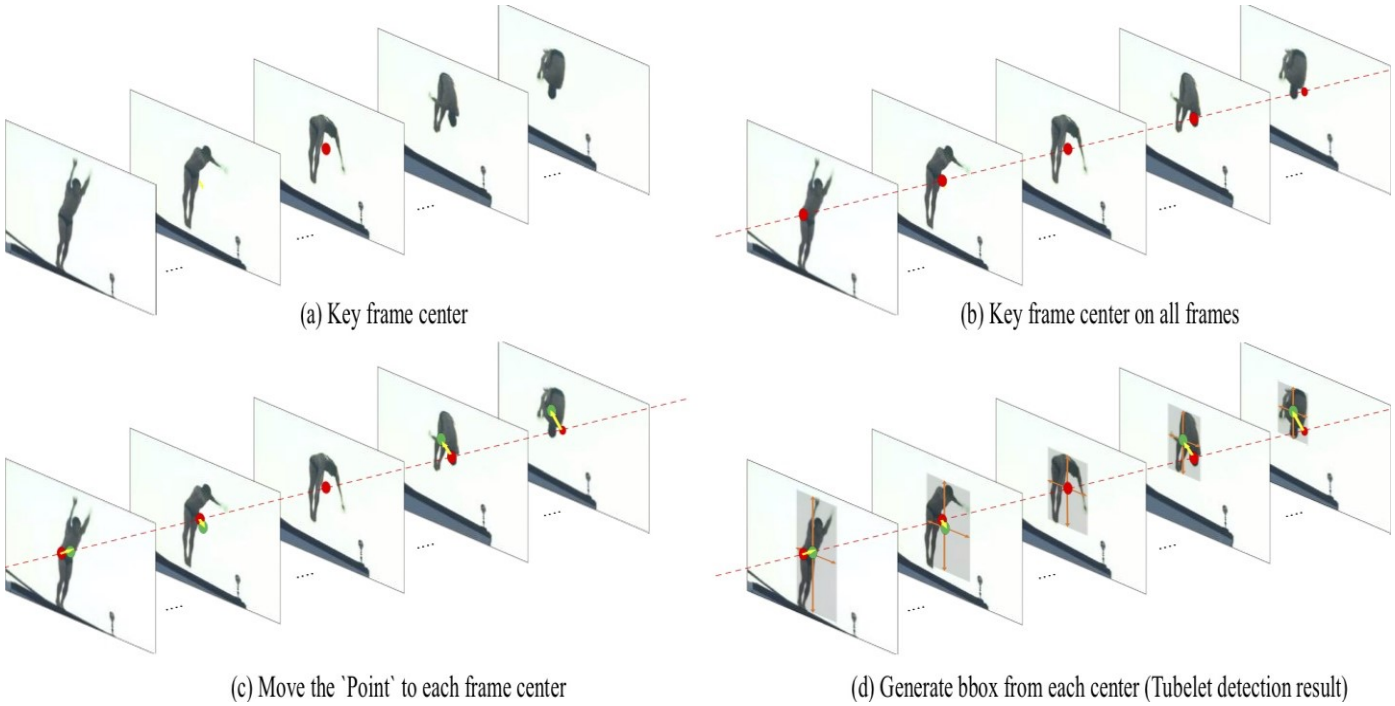
- We analyze the drawback of RoI-Pooling based pipeline in action detection framework.
- We revisit RCNN-like pipeline for action detection
- Context-Aware RCNN: a baseline method
- Thorough ablation experiments provide powerful support for our insights

Tubelet-level Action Detection



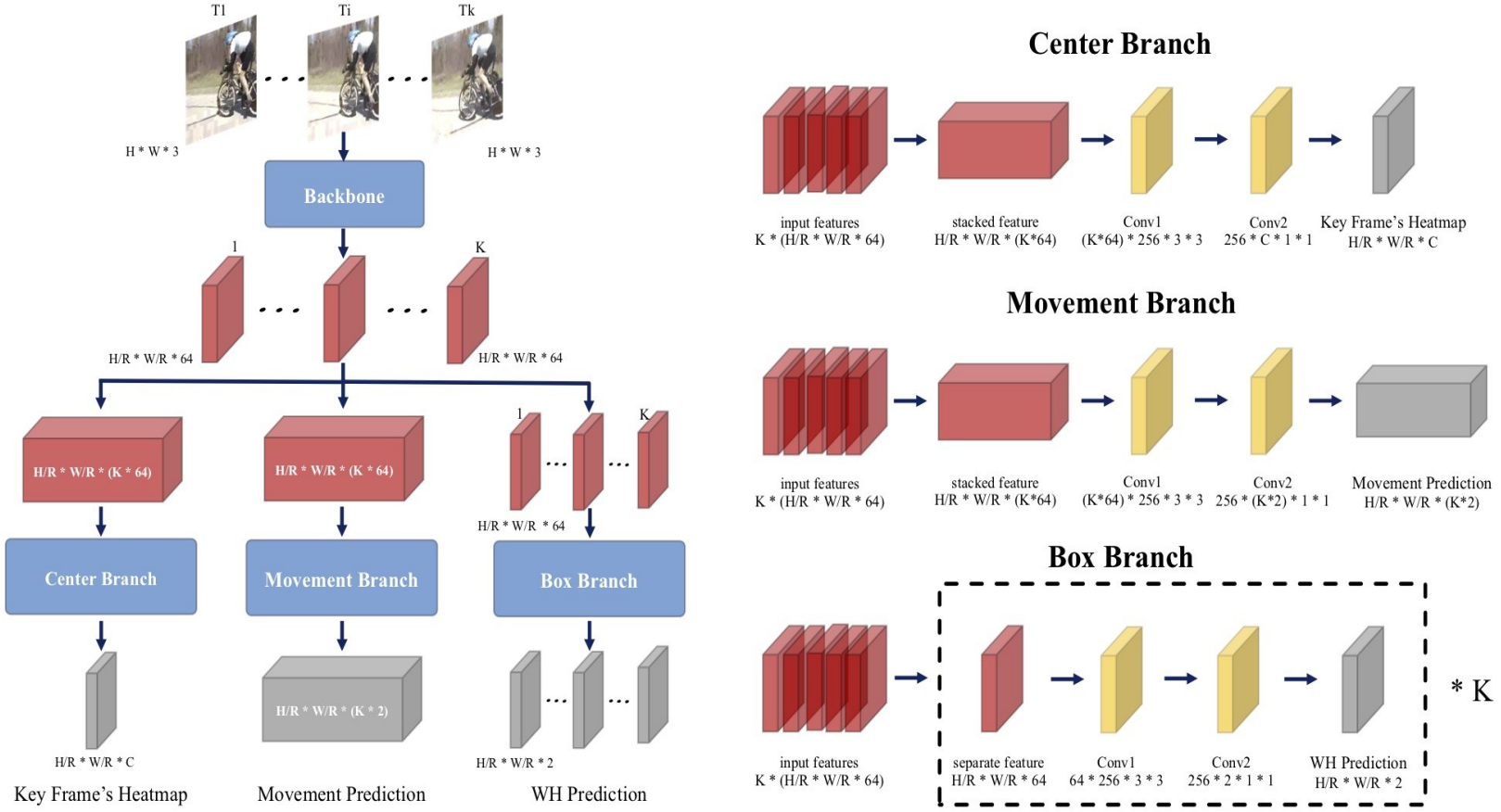
From Frame-Level detection to Tubelet Detection

Motivation



Simplify each action instance as a trajectory of moving points

Overview



A single stage anchor free Tubelet detection pipeline

Technical details

- 2D backbones for frame wise feature extraction + Customized detection heads
- **Center branch:** detect instance center at key frame

$$\ell_{\text{center}} = -\frac{1}{n}\sum_{x,y,c}\begin{cases}(1-\hat{L}_{xyc})^{\alpha}\log(\hat{L}_{xyc}) & \text{if } L_{xyc}=1\\ (1-L_{xyc})^{\beta}(\hat{L}_{xyc})^{\alpha}\log(1-\hat{L}_{xyc}) & \text{otherwise}\end{cases} \quad (1)$$

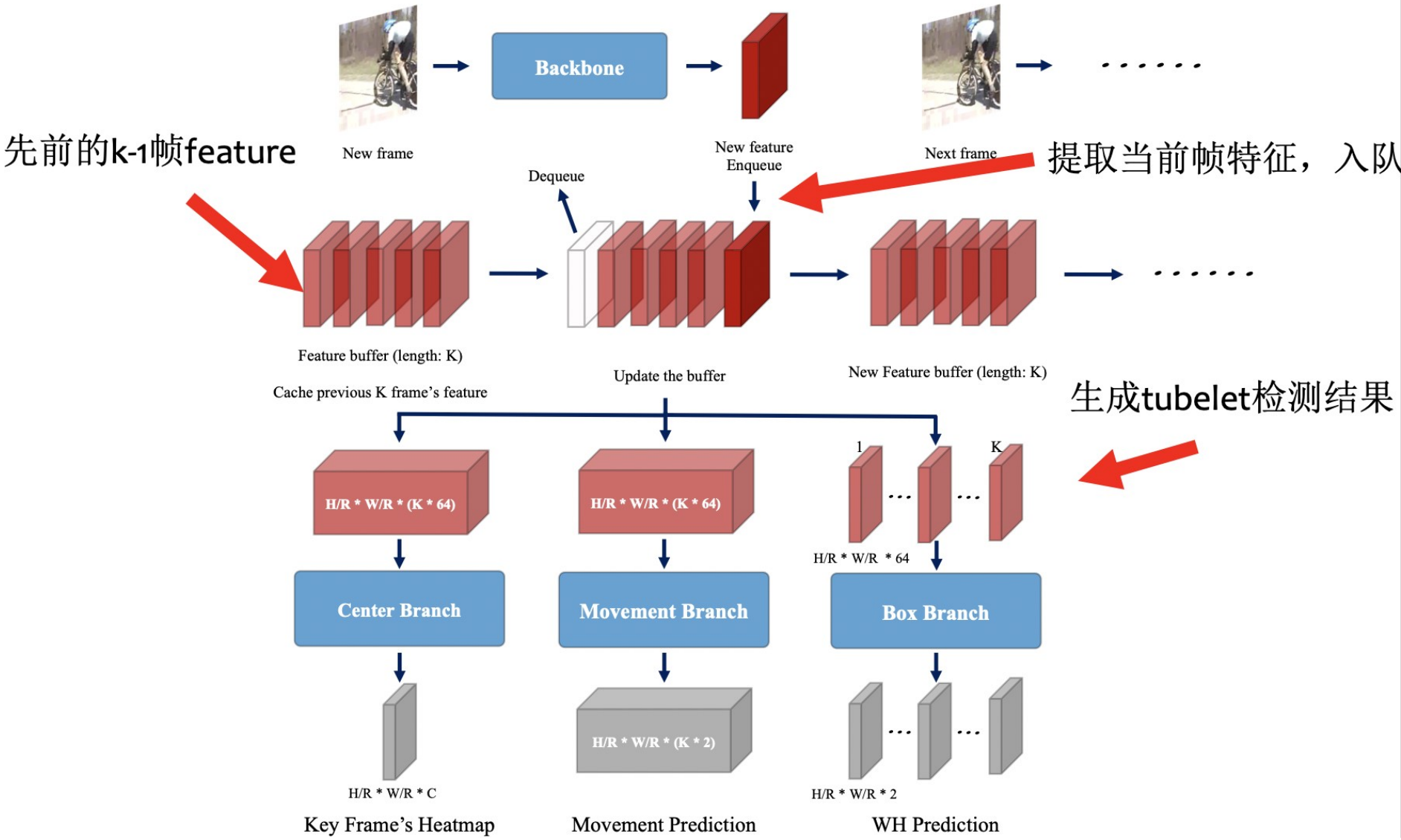
- **Movement branch:** move center temporally

$$\ell_{\text{movement}} = \frac{1}{n}\sum_{i=1}^{n}|\hat{M}_{x_{key_i},y_{key_i}} - m_i|. \quad (5)$$

- **Box branch:** determine spatial extent at each detected center

$$\ell_{\text{box}} = \frac{1}{n}\sum_{i=1}^{n}\sum_{j=1}^{K}|\hat{S}^{j}_{p_i^j} - s_i^j|. \quad (8)$$

Online testing



Ablation Study

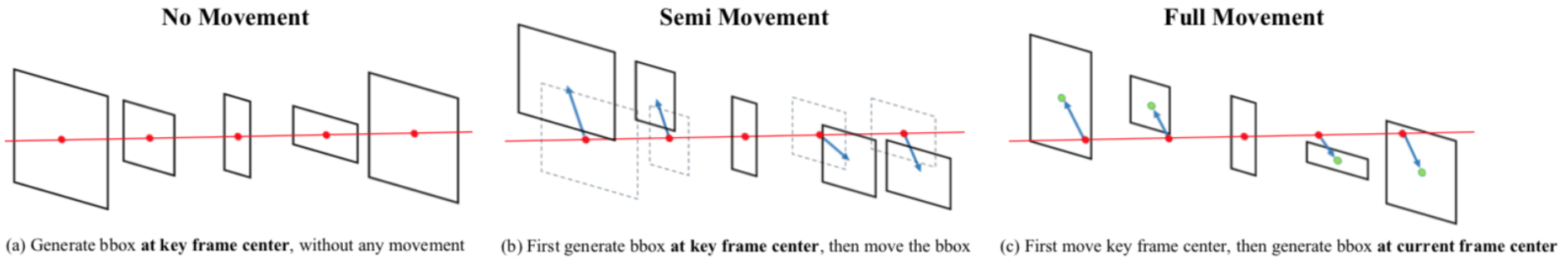


Fig. 3. Illustration of Three Movement Strategies. Note that the arrow represents moving according to Movement Branch prediction, the red dot represents the key frame center and the green dot represents current frame center, which is localized by moving key frame center according to Movement Branch prediction.

Table 1. Exploration study on MOC detector design with various combinations of movement strategies on UCF101-24.

Method	Strategy		F-mAP@0.5(%)	Video-mAP(%)			
	Move Center	Bbox Align		@0.2	@0.5	@0.75	0.5:0.95
No Movement			68.22	68.91	37.77	19.94	19.27
Semi Movement	✓		69.78	76.63	48.82	**27.05**	26.09
Full Movement(MOC)	✓	✓	**71.63**	**77.74**	**49.55**	27.04	**26.09**

Ablation Study

Method	F-mAP@0.5(%)	Video-mAP(%)			
		@0.2	@0.5	@0.75	0.5:0.95
Flow Guided Movement	69.38	75.17	42.28	22.26	21.16
Cost Volume Movement	69.63	72.56	43.67	21.68	22.46
Accumulated Movement	69.40	75.03	46.19	24.67	23.80
Center Movement	**71.63**	**77.74**	**49.55**	**27.04**	**26.09**

Tubelet Duration	F-mAP@0.5(%)	Video-mAP(%)			
		@0.2	@0.5	@0.75	0.5:0.95
$K = 1$	68.33	65.47	31.50	15.12	15.54
$K = 3$	69.94	75.83	45.94	24.94	23.84
$K = 5$	71.63	77.74	49.55	27.04	26.09
$K = 7$	**73.14**	**78.81**	**51.02**	**27.05**	**26.51**
$K = 9$	72.17	77.94	50.16	26.26	26.07

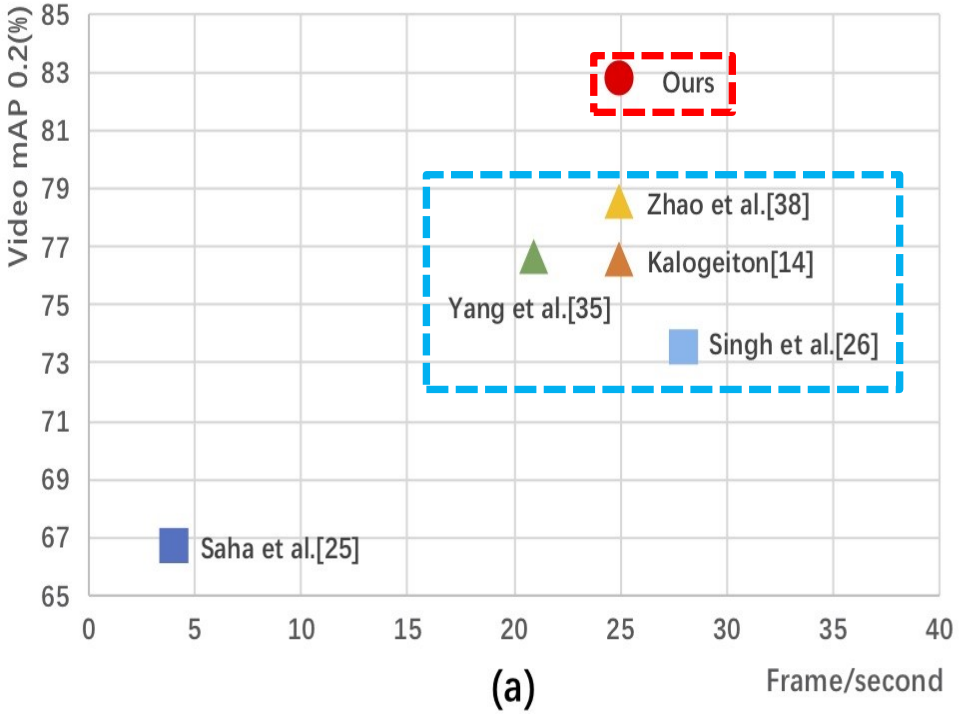
Compare with SOTA

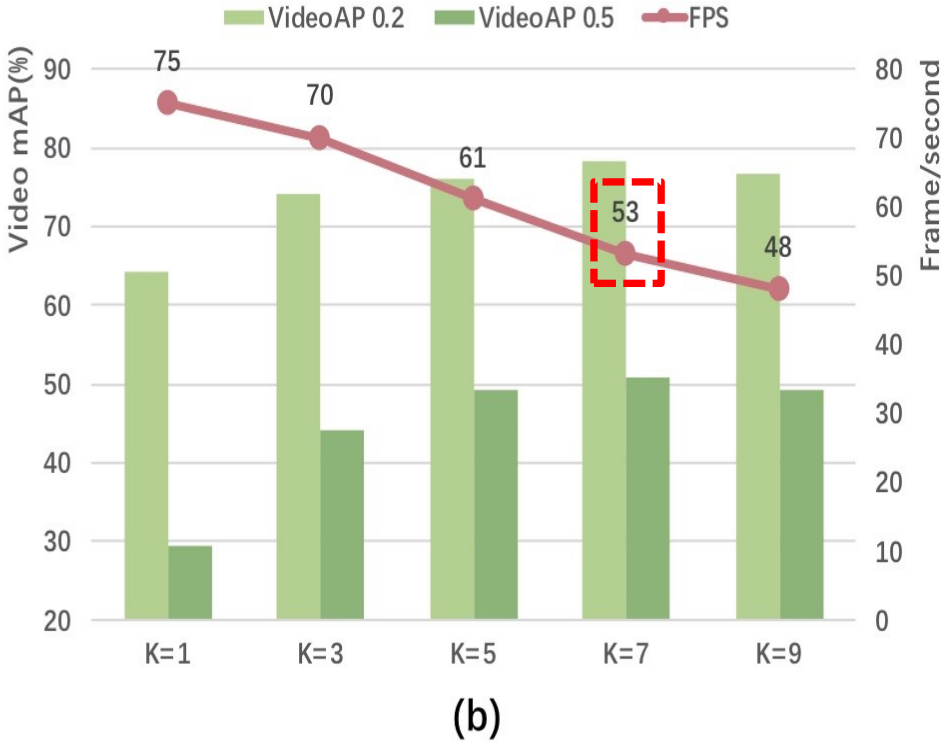
MULTIMEDIA COMPUTING GROUP
媒体计算研究组

Table 4. Comparison with the state of the art on JHMDB (trimmed) and UCF101-24 (untrimmed). Ours (MOC) with † is pretrained on ImageNet and the other is pretrained on COCO.

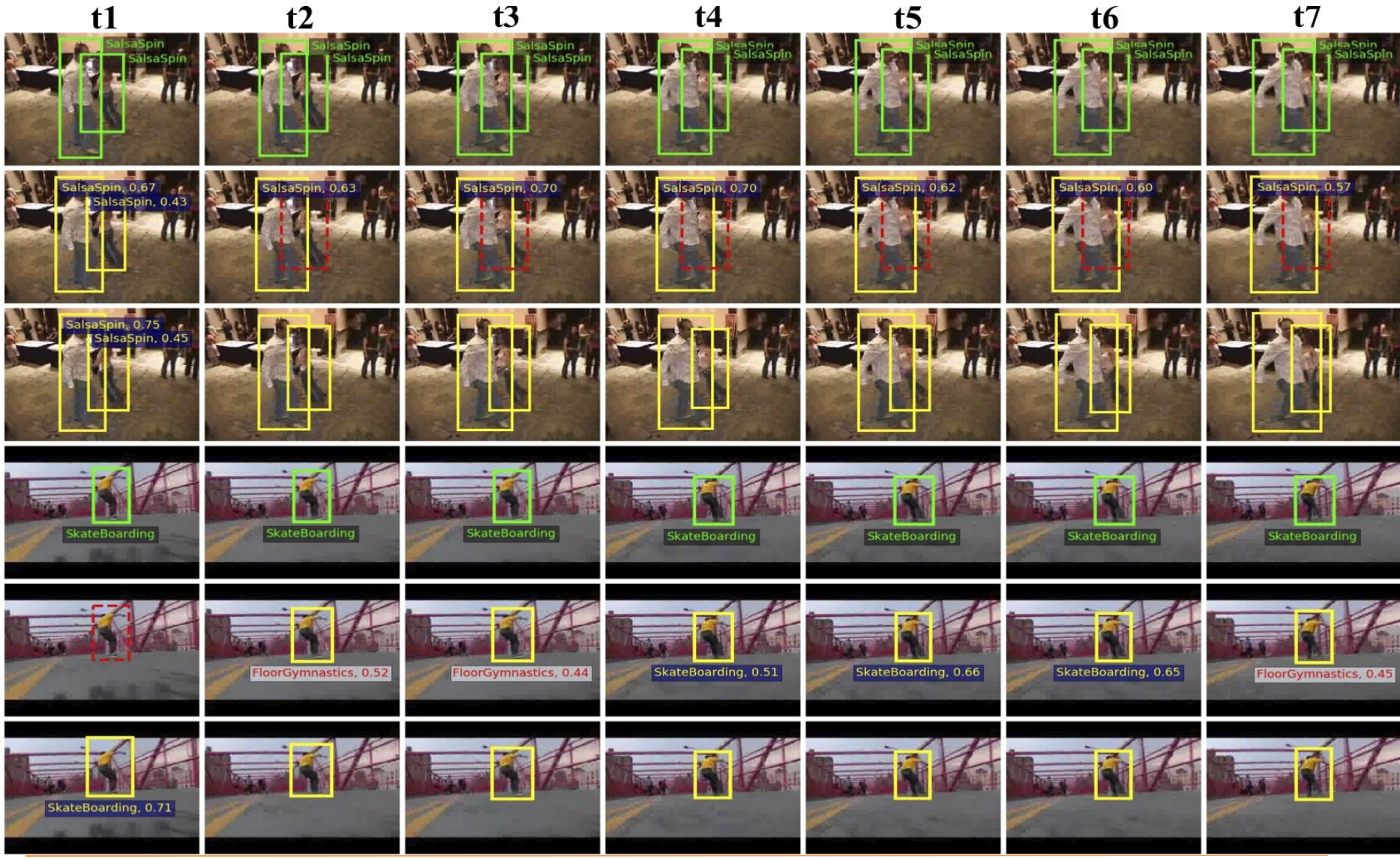
Method	JHMDB					UCF101-24				
	Frame-mAP@0.5(%)	Video-mAP (%)				Frame-mAP@0.5 (%)	Video-mAP (%)			
		@0.2	@0.5	@0.75	0.5:0.95		@0.2	@0.5	@0.75	0.5:0.95
2D Backbone										
Saha *et al.* 2016 [23]	-	72.6	71.5	43.3	40.0	-	66.7	35.9	7.9	14.4
Peng *et al.* 2016 [19]	58.5	74.3	73.1	-	-	39.9	42.3	-	-	-
Singh *et al.* 2017 [24]	-	73.8	72.0	44.5	41.6	-	73.5	46.3	15.0	20.4
Kalogeiton *et al.* 2017 [12]	65.7	74.2	73.7	52.1	44.8	69.5	76.5	49.2	19.7	23.4
Yang *et al.* 2019 [32]	-	-	-	-	-	75.0	76.6	-	-	-
Song *et al.* 2019 [25]	65.5	74.1	73.4	52.5	44.8	72.1	77.5	52.9	21.8	24.1
Zhao *et al.* 2019 [35]	-	-	74.7	53.3	45.0	-	78.5	50.3	22.2	24.5
Ours (MOC)†	68.0	76.2	75.4	68.5	54.0	76.9	81.3	**54.4**	29.5	**28.4**
Ours (MOC)	**70.8**	**77.3**	**77.2**	**71.7**	**59.1**	**78.0**	**82.8**	53.8	**29.6**	28.3
3D Backbone										
Hou *et al.* 2017 [9](C3D)	61.3	**78.4**	76.9	-	-	41.4	47.1	-	-	-
Gu *et al.* 2018 [7] (I3D)	73.3	-	78.6	-	-	**76.3**	-	**59.9**	-	-
Sun *et al.* 2018 [27](S3D-G)	**77.9**	-	**80.1**	-	-	-	-	-	-	-

Runtime Analysis

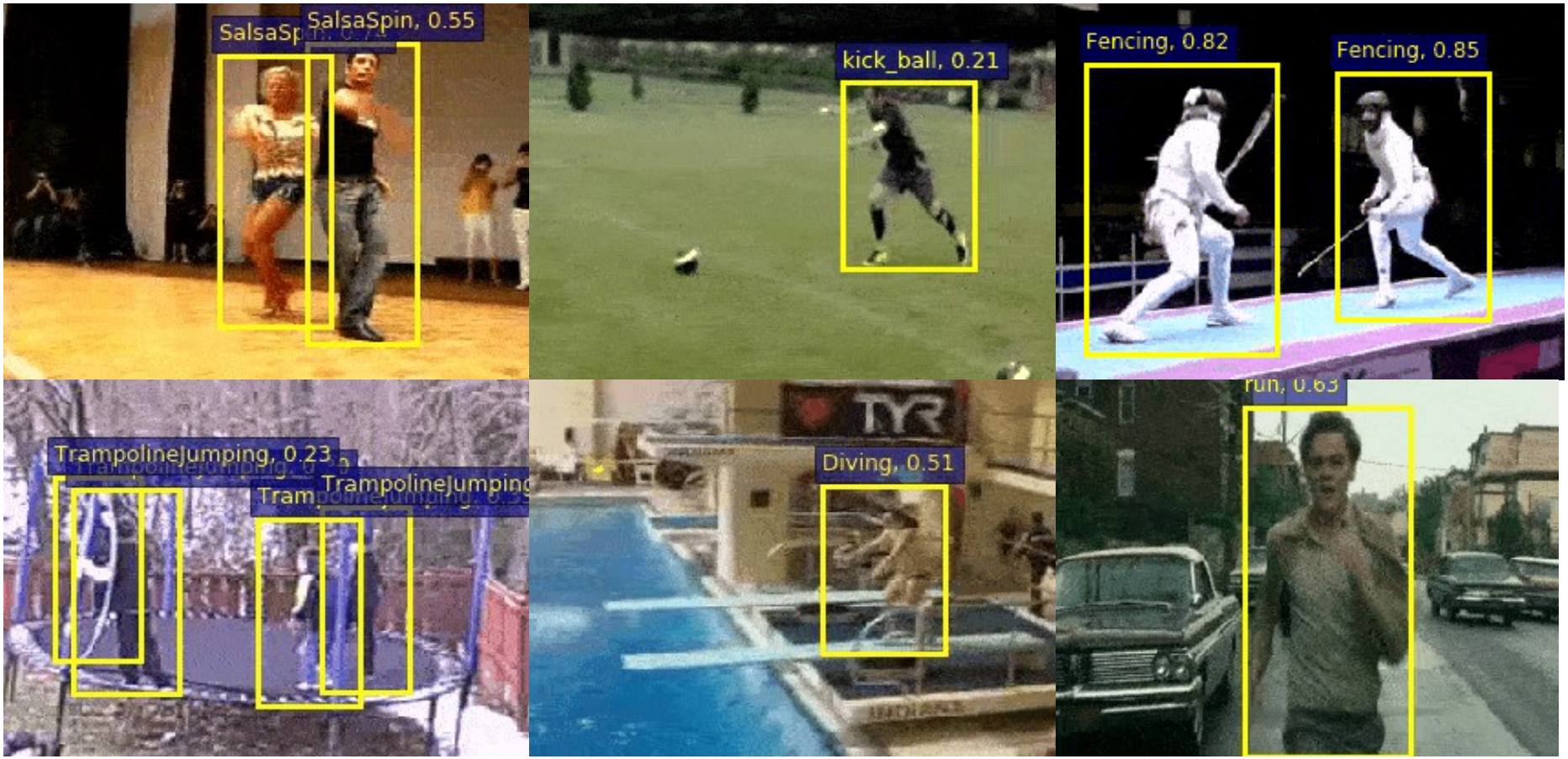




Visualization



Visualization



Conclusion

- MOC is conceptually simple, computationally efficient, and more precise tubelet detector.
- Anchor free is possible for tubelet detection
- Study on different of forms for offset regression
- SOTA performance on JHMDB and UCF24, in particular high-IoU video mAP.

Outline

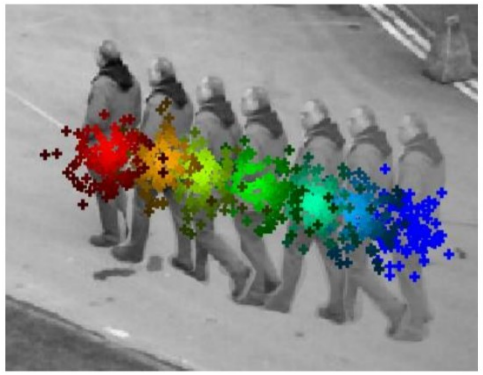
MULTIMEDIA COMPUTING GROUP
媒体计算研究组

- **视频表征模型**
  - TEINet: Towards an Efficient Architecture for Video Recognition (AAAI 2020)
  - TAM: Temporal Adaptive Module for Video Recognition (arXiv 2020)
- **视频检测框架**
  - Context-Aware RCNN：a Baseline for Action Detection in Videos (ECCV 2020)
  - Actions as Moving Points (ECCV 2020)
- **视频跟踪框架**
  - Fully convolutional online tracking (arXiv 2020)

Video Object Tracking

- **Tracking** is to estimate the number and state of objects in a region of interest.

- Challenge: appearance variation, background clutter, occlusion, deformation.



From Frame-Level or short-term detection to long-term detection

Motivation

- Online learning is effective to handle object appearance variation.
- Current tracking: classification branch + regression branch
- Online learning is hard to apply to regression branch due to the design complexity in regression scheme.

A fully convolutional online tracker

Method

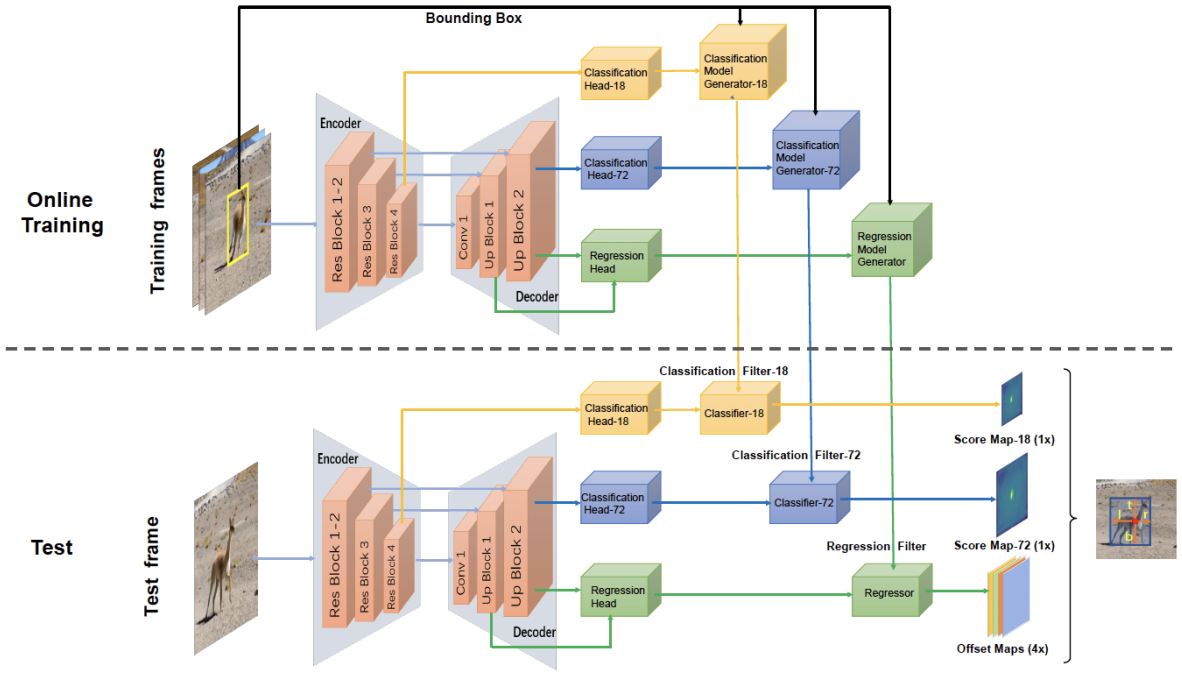


Figure 2: **Fully Convolutional Online Tracker**. Our FCOT presents a fully convolutional framework for online tracking, which is composed of an encoder-decoder backbone, two classification heads and a regression head on top for task-specific feature extraction, classification and regression model generators, classifiers and regressor. Our FCOT follows a simple online tracking recipe, where both classification and regression model generators will produce a target-specific classifier and regressor weight based on an online updated training set. These weights will be adaptive for each tracked object, thus making our FCOT more accurate and robust. Details of regression model generator could be found in Figure 3.

Regression Model Generator

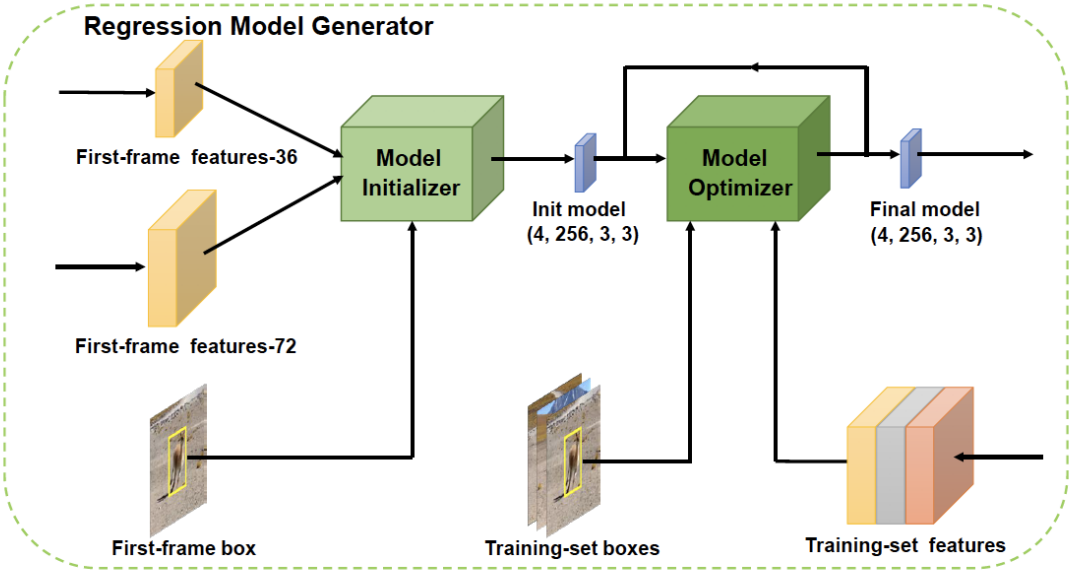


Figure 3: Our proposed **Regression Model Generator** produces the regressor weights in regression branch. It consists of a model initializer and a model optimizer. The model initializer takes the features extracted from the decoder and bounding box of the first frame as input and generates the initial model. The features extracted from the regression head and bounding boxes on training set are then fed into the model optimizer to optimize the regression model iteratively.

Multi-scale Classification

- In general, in visual tracking, score map with coarse resolution produces robust yet not accurate results, while a high resolution map is with a complementary property. Thus, we devise a multi-scale prediction strategy for classification branch to handle the issue of similar object confusion, and also improve accuracy.

$$M_{cls}^{(18)} = \phi^{(18)}(\omega(I_t)) * f_{cls}^{(18)}, \; M_{cls}^{(72)} = \phi^{(72)}(\omega(I_t)) * f_{cls}^{(72)}, \; M_{cls} = \alpha M_{cls}^{(18)} + \beta M_{cls}^{(72)}, \quad (1)$$

where the parameter ω denotes the weights of an encoder-decoder backbone, $\phi^{(18)}$ is the classification head of Score Map-18 and $\phi^{(72)}$ of Score Map-72, I_t represents for the test frame, $f_{cls}^{(18)}$ and $f_{cls}^{(72)}$ are the classification models of each branch generated by the corresponding model generators, α and β are weights of the two score maps. During training, the classification target is a Gaussian function map centered at the ground-truth target center c_t.

Ablation Study

Table 1: Ablation analysis of multi-scale prediction in classification branch on the VOT2018 and TrackingNet datasets. Score-18 represents for using Score Map-18 for classification and Score-72 for Score Map-72. The best results are highlighted by **bold**.

Score-18	Score-72	VOT2018			TrackingNet		
		EAO	Rob.	Acc.	P_{norm}	Prec.	Succ.
✓		0.435	0.155	0.547	0.801	0.681	0.728
	✓	0.399	0.211	**0.610**	0.817	0.714	0.745
✓	✓	**0.508**	**0.108**	0.600	**0.828**	**0.723**	**0.751**

Table 2: Ablation analysis of online regression model on the VOT2018 and TrackingNet datasets. The best results are highlighted by **bold**.

	VOT2018			TrackingNet		
	EAO	Rob.	Acc.	P_{norm}	Prec.	Succ.
W/O Online	0.447	0.159	0.603	0.824	0.719	0.748
W/ Online	**0.508**	**0.108**	0.600	**0.828**	**0.723**	**0.751**

Table 3: Ablation analysis of feature fusion in regression head on the VOT2018 and TrackingNet datasets. UP-1 represents for using features of the first up-sample layer in decoder and UP-2 for features of the second up-sample layer in decoder. The best results are highlighted by **bold**.

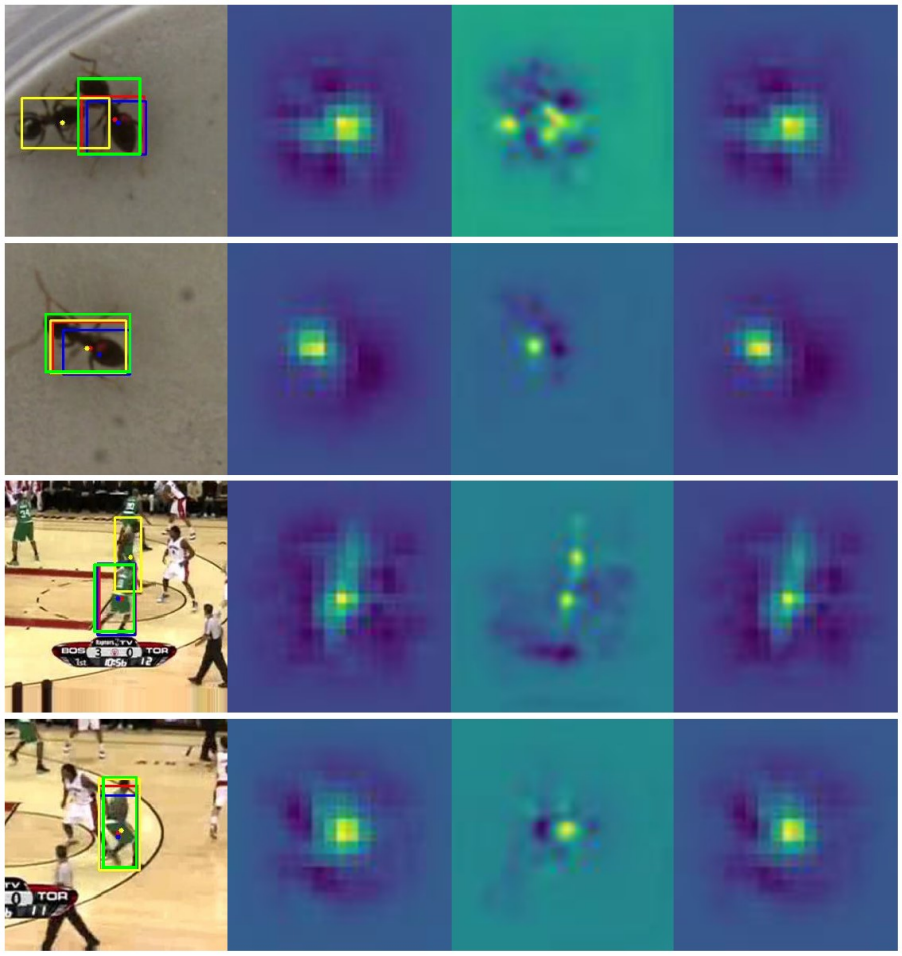
UP-1	UP-2	VOT2018			TrackingNet		
		EAO	Rob.	Acc.	P_{norm}	Prec.	Succ.
✓		0.400	0.173	0.590	0.826	0.716	0.743
	✓	0.420	0.155	**0.606**	0.826	0.719	0.749
✓	✓	**0.508**	**0.108**	0.600	**0.828**	**0.723**	**0.751**

Comparison

Table 4: Results on several benchmarks. The best two results are highlighted by **red bold** and **blue bold**.

	VOT2018			TrackingNet			GOT-10k			UAV123		NFS	
	EAO	Rob.	Acc.	P_{norm}	Prec.	Succ.	AO	$SR_{0.5}$	$SR_{0.75}$	Succ.	Prec.	Succ.	Prec.
MDNet [27]	-	-	-	0.705	0.565	0.606	0.299	0.303	0.099	0.528	-	0.422	-
ECO [4]	0.280	0.276	0.484	0.618	0.492	0.554	0.316	0.309	0.111	0.525	0.741	0.466	-
SiamFC [1]	0.188	0.585	0.503	0.652	0.518	0.559	0.348	0.353	0.098	-	-	-	-
DaSiamRPN [39]	0.383	0.276	0.586	0.733	0.591	0.618	-	-	-	0.586	0.796	-	-
SiamRPN++ [19]	0.414	0.234	**0.600**	0.800	0.694	0.733	-	-	-	0.613	0.807	-	-
ATOM [5]	0.401	0.204	0.590	0.771	0.648	0.703	0.556	0.634	0.402	0.632	0.844	0.580	0.700
DiMP [2]	**0.440**	**0.153**	0.597	**0.801**	0.687	0.740	**0.611**	**0.717**	**0.492**	**0.643**	**0.849**	**0.615**	**0.741**
SiamFC++ [34]	0.426	0.183	0.587	0.800	**0.705**	**0.754**	0.595	0.695	0.479	-	-	-	-
FCOT	**0.508**	**0.108**	**0.600**	**0.828**	**0.723**	**0.751**	**0.640**	**0.763**	**0.517**	**0.654**	**0.875**	**0.632**	**0.761**

Visualization of score maps



We can see from **the first and the third row** that the results of just using Score Map-72 are deviated from the ground truth while trackers that using just Score Map-18 and using both of them can discriminate the positive object from the similar ones. It demonstrates that Score Map-18 is helpful for the robustness of the tracker.

While from **the second and the last row**, we can derive that the predicted bounding boxes and centers of using Score Map-72 are more precise than only using Score Map-18.

In consequence, multi-scale classification strategy is helpful for both the robustness and accuracy.

Results

Results

Conclusion

- FCOT is simple and effective fully convolutional online tracker
- Online learning is useful for both classification and regression branches
- Multi-scale classification is able to handle similar objects
- SOTA performance on several benchmarks.

总结：表征、检测、跟踪

MULTIMEDIA COMPUTING GROUP
媒体计算研究组

- 表征：有效的时序运动建模
  - Self-attention & Dynamic Filtering （自适应）
- 检测：空间信息和短时时序信息
  - Crop vs. RoI pooling (空间信息更富)
  - Anchor free tubelet detection（简洁&短时运动）
- 跟踪：长时时序的变化
  - Online learning （适应时序变化）
  - Target guided anchor free regression （简洁 & 精确）

主页：http://mcg.nju.edu.cn/

代码：https://github.com/MCG-NJU/

谢谢大家！

2020/11/10

MCG MULTIMEDIA COMPUTING GROUP 媒体计算研究组