Learning Transferable Self-attentive Representations for Action Recognition in Untrimmed Videos with Weak Supervision

Xiao-Yu Zhang[1] Haichao Shi[1,2] Changsheng Li[3]
Xiaobin Zhu[4] Lixin Duan[3] Kai Zheng[3]

[1]Institute of Information Engineering, Chinese Academy of Sciences

[2]School of Cyber Security, University of Chinese Academy of Sciences

[3]University of Electronic Science and Technology of China

[4]Beijing Technology and Business University

AAAI, 2019
Presenter: Haichao Shi

Outline

- 1. Introduction
  - Motivation
  - Overview
  - Related Work
- 2. Method (TSRNet)
  - Overview
  - Learning
  - Localization
- 3. Evaluation
  - Settings
  - Accuracy
  - Analysis

Outline

- 1. Introduction
    - Motivation
    - Overview
    - Related Work
- 2. Method (TSRNet)
    - Overview
    - Learning
    - Localization
- 3. Evaluation
    - Settings
    - Accuracy
    - Analysis

Motivation

- 1. It may be difficult to adapt to **large-scale action recognition** in more realistic and challenging scenario only relying on the **existing precise temporal annotations**.
- 2. There might even be **no sensible definition** about the exact temporal extent of actions and these temporal annotations may be **subjective** and **not consistent** across different persons, so weakly supervised learning is an effective way to perform the task.

Motivation

- 3. Current action recognition methods heavily rely on trimmed videos for model training, while it is **expensive** and **time-consuming** to acquire a large-scale trimmed video dataset.
- 4. The breakthrough made by **self-attention** on computer vision tasks makes it possible to **weight key frames** to eliminate the influence of background frames.
- 5. The abundant and useful information contained in trimmed videos contribute the use of **transfer learning**.

Outline

- 1. Introduction
  - Motivation
  - Overview
  - Related Work
- 2. Method (TSRNet)
  - Overview
  - Learning
  - Localization
- 3. Evaluation
  - Settings
  - Accuracy
  - Analysis

Overview

- Different data distributions in the source and target domain, represented as trimmed videos and untrimmed videos.

In this paper, only video-level labels are provided in untrimmed videos, simultaneously publicly available trimmed videos are leveraged as additional information to learn a robust model.

- Method:

A **self-attention module** for each domain: capture specific domain properties.

A **transfer module**: capture representations shared by domains.

Outline

- 1. Introduction
  - Motivation
  - Overview
  - Related Work
- 2. Method (TSRNet)
  - Overview
  - Learning
  - Localization
- 3. Evaluation
  - Settings
  - Accuracy
  - Analysis

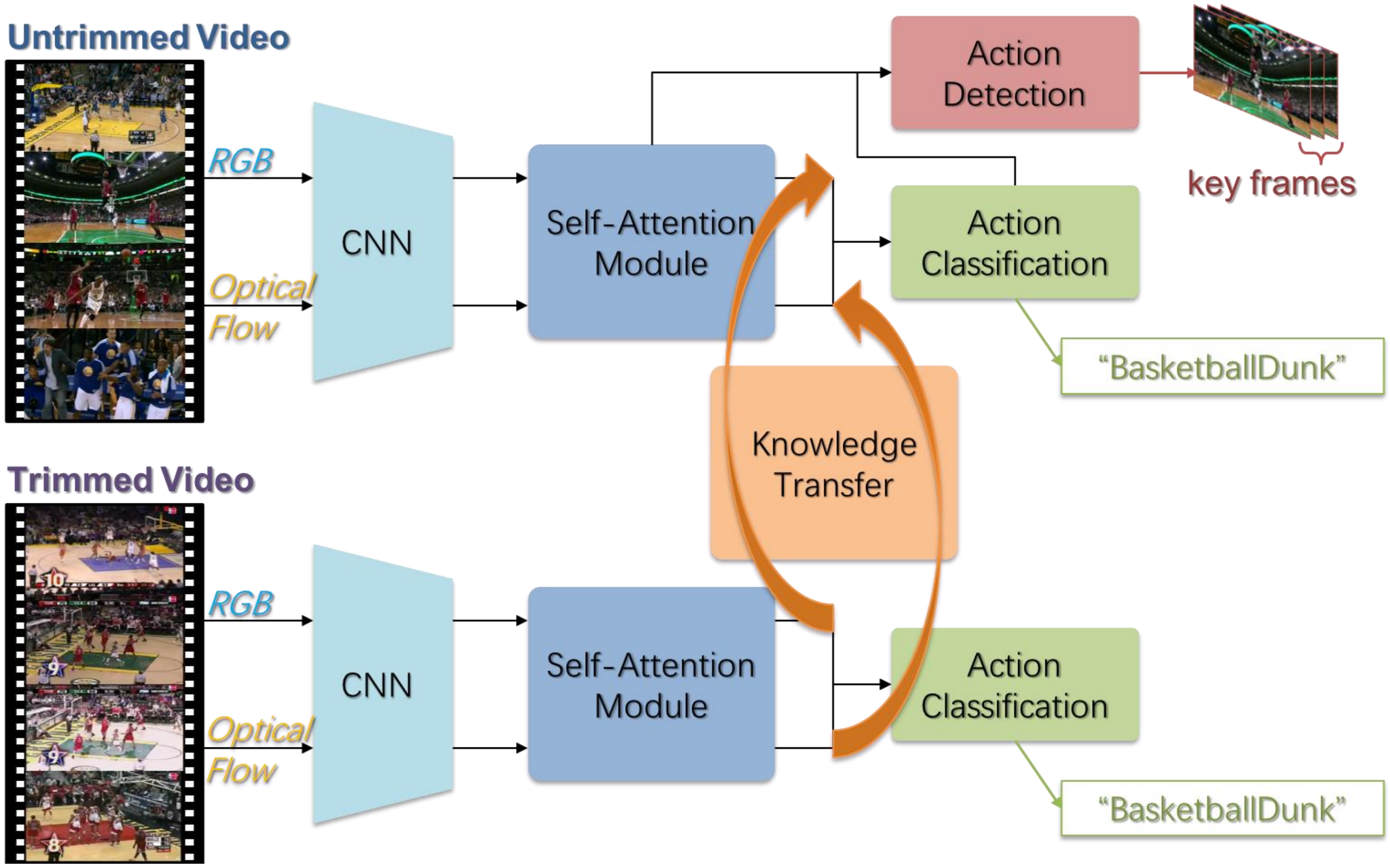
Related Work

- Action Recognition
  - Two-Stream Network, 3D Convolutional neural networks(C3D), Temporal Segment Network(TSN)
- Temporal Action Detection
  - Fully-supervised: S-CNN, SSN; Weakly-supervised: UntrimmedNet, STPN, W-TALC
- Transfer Learning
  - Maximum Mean Discrepancy(MMD), JAN, DAN
- Attention Mechanism
  - Self-attention

Outline

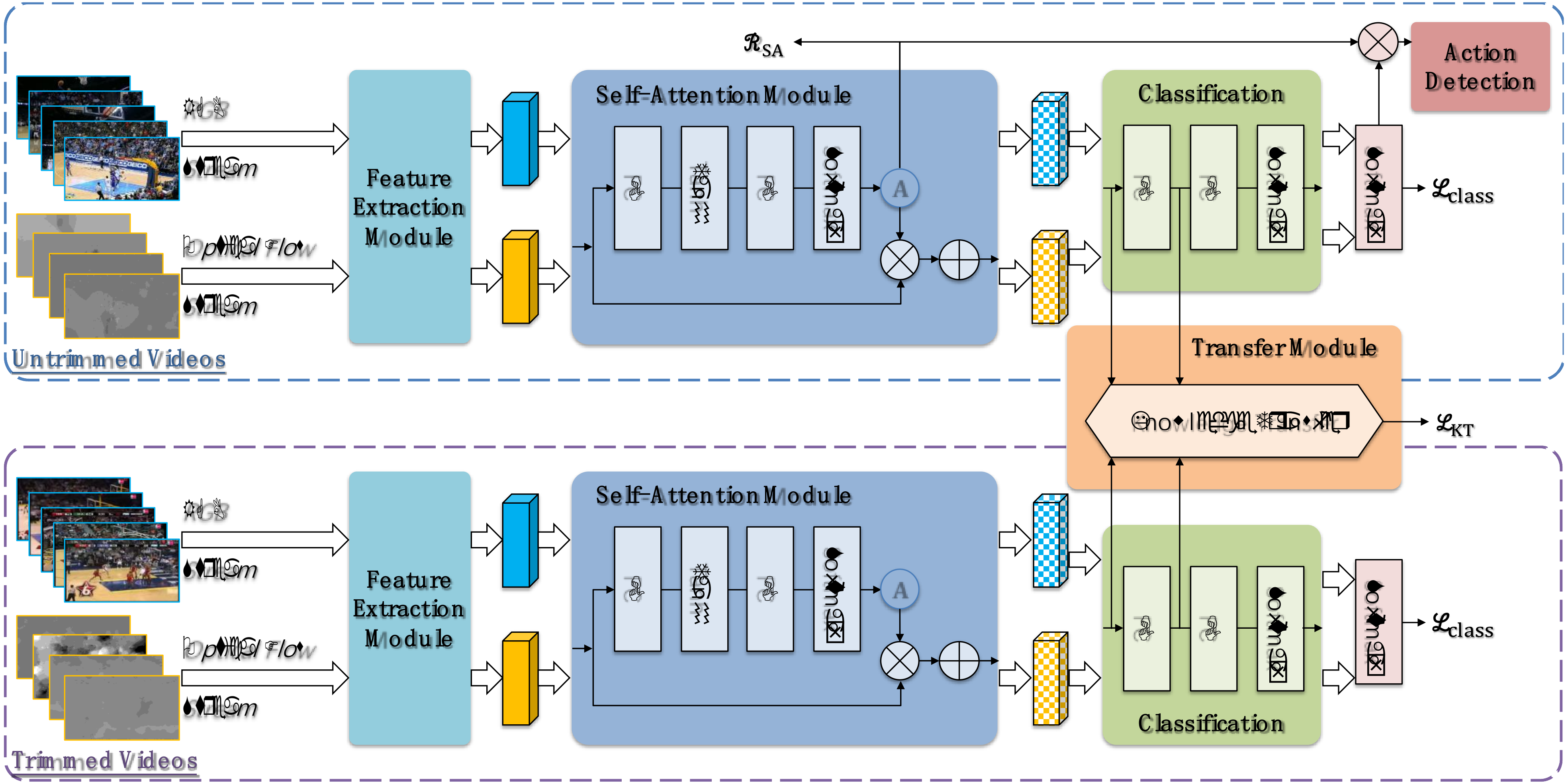
- 1. Introduction
  - Motivation
  - Overview
  - Related Work
- 2. Method (TSRNet)
  - Overview
  - Learning
  - Localization
- 3. Evaluation
  - Settings
  - Accuracy
  - Analysis

Overview



- **TSRNet first extracts high-level semantic features of frames and optical flows.**
- **Then it localizes the action frames utilizing two separate self-attention modules.**
- **Later, the knowledge extracted from the trimmed training videos is transferred to enhance the classification performance of the overall model for the untrimmed videos.**

Overview



A: the attention matrix

Outline

- 1. Introduction
  - Motivation
  - Overview
  - Related Work
- 2. Method (TSRNet)
  - Overview
  - Learning
  - Localization
- 3. Evaluation
  - Settings
  - Accuracy
  - Analysis

Learning

Self-attentive Action Classification

$$\mathbf{m} = \mathbf{Xa} = \mathbf{X}(\mathrm{softmax}(\mathbf{w_2} \cdot \tanh(\mathbf{W_1X})))^{\mathrm{T}}$$

$$\mathbf{M} = \mathbf{XA} = \mathbf{X}(\mathrm{softmax}(\mathbf{W_2} \cdot \tanh(\mathbf{W_1X})))^{\mathrm{T}}$$

$$\mathcal{L}_{SA} = \mathcal{L}_{class} + \mathcal{R}_{SA}$$

$$\mathcal{R}_{SA} = \alpha\mathcal{R}_{smooth} + \beta\mathcal{R}_{diversity} + \gamma\mathcal{R}_{sparsity}$$

$$\mathcal{R}_{smooth} = \sum_{i=1}^{n-1}(a_i - a_{i+1})^2 \,, \mathcal{R}_{diversity} = ||\mathrm{A^TA} - \mathrm{I}||_{\mathrm{F}}^2 \,, \mathcal{R}_{sparsity} = ||\mathrm{A^T}||_{2,1}$$

$\mathcal{L}_{class}$: the standard multi-label cross-entropy loss

a: attention weights vector, $\mathbf{A}$: attention weights matrix

$\mathbf{X}$: feature matrix, $\mathbf{m}$: a weighted sum of feature vectors, $\mathbf{M}$: self-attentive feature matrix

Learning

Knowledge Transfer between Trimmed and Untrimmed Videos

$\mathcal{L}_{KT} = \mathcal{L}_{FC1} + \mathcal{L}_{FC2}$

$\mathcal{L}_{FC1} = MMD^2(\mathcal{T}, \mathcal{U})$

$$= \frac{1}{n_T^2} \sum_{i=1}^{n_T} \sum_{j=1}^{n_T} k(t_i, t_j) + \frac{1}{n_U^2} \sum_{i=1}^{n_U} \sum_{j=1}^{n_U} k\left(u_i, u_j\right) - \frac{2}{n_T \cdot n_U} \sum_{i=1}^{n_T} \sum_{j=1}^{n_U} k(t_i, u_j)$$

$\mathcal{L}_{FC2} = MMD^2(FC1(\mathcal{T}), FC1(\mathcal{U}))$

$\mathcal{T} = \{\boldsymbol{t}_i|_{\mathrm{i}=1}^{n_T}\}$, $\mathcal{U} = \{\boldsymbol{u}_i|_{\mathrm{i}=1}^{n_U}\}$, represent the sets of trimmed and untrimmed videos features.

$k(\cdot,\cdot)$: the predefined Gaussian kernel function.

Total Loss: $\boldsymbol{\mathcal{L}} = \boldsymbol{\mathcal{L}_{SA}} + \boldsymbol{\mathcal{L}_{KT}}$

Outline

- 1. Introduction
  - Motivation
  - Overview
  - Related Work
- 2. Method (TSRNet)
  - Overview
  - Learning
  - Localization
- 3. Evaluation
  - Settings
  - Accuracy
  - Analysis

Localization

$$w_i^c = \bar{a}_i s_c$$

$$\bar{a}_i = \frac{1}{r}\sum\nolimits_{j=1}^{r} \mathbf{A}_{ij}$$

$$\bar{w}_i^c = \theta \cdot w_{i,RGB}^c + (1-\theta) \cdot w_{i,Flow}^c$$

$$t_{start} = \frac{ind_{start}}{F}$$

$$t_{end} = \frac{ind_{end}}{F}$$

w_i^c: the weighted score of each frame i for class c.

s_c: $s_c = [s_1, s_2, \dots, s_m]^T \in \mathbb{R}^{m\times 1}$ is the output of softmax layer.

$[ind_{start}, ind_{end}]$: the frames indices of starting and ending positions.

F: the fps(frames per second) of videos.

Outline

- 1. Introduction
  - Motivation
  - Overview
  - Related Work
- 2. Method (TSRNet)
  - Overview
  - Learning
  - Localization
- 3. Evaluation
  - Settings
  - Accuracy
  - Analysis

Settings

Evaluation is based on training on the paired datasets.

- Data for training:

  Source domain training: Trimmed videos from the source domain.

  Domain adaptation training: Untrimmed videos from the target domain.

- Test:

  Test set from the target domain

- Transfer scenarios:

  (a). UCF101 to THUMOS14

  (b). UCF101 to ActivityNet1.3

Outline

- 1. Introduction
  - Motivation
  - Overview
  - Related Work
- 2. Method (TSRNet)
  - Overview
  - Learning
  - Localization
- 3. Evaluation
  - Settings
  - Accuracy
  - Analysis

Accuracy

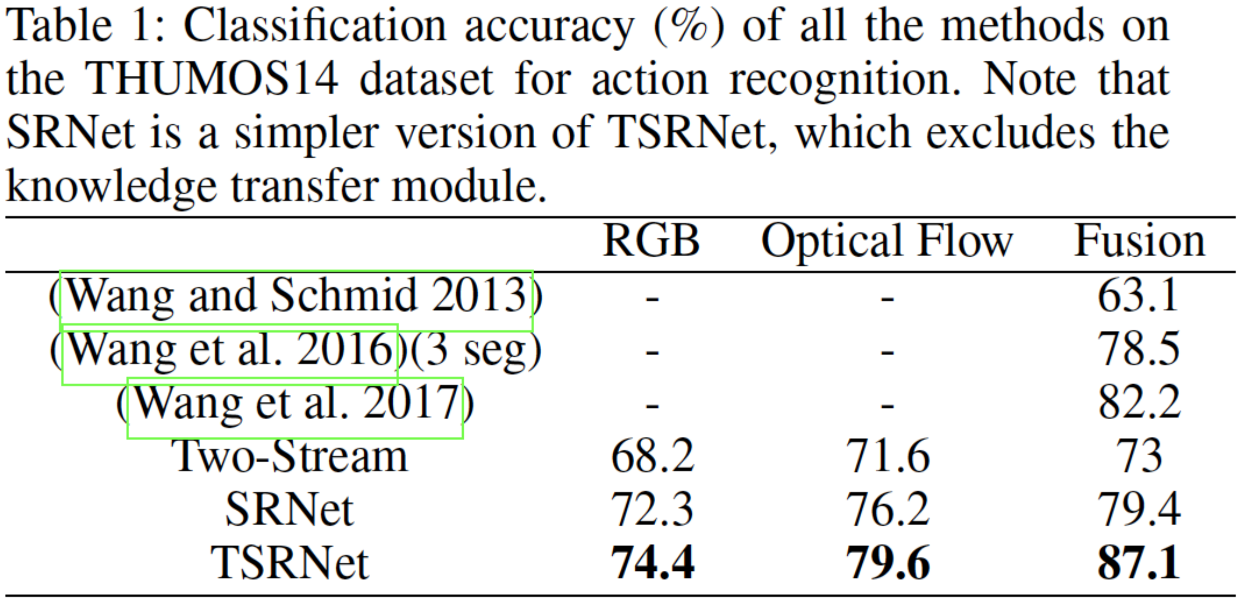


Table 1: Classification accuracy (%) of all the methods on the THUMOS14 dataset for action recognition. Note that SRNet is a simpler version of TSRNet, which excludes the knowledge transfer module.

	RGB	Optical Flow	Fusion
(Wang and Schmid 2013)	-	-	63.1
(Wang et al. 2016)(3 seg)	-	-	78.5
(Wang et al. 2017)	-	-	82.2
Two-Stream	68.2	71.6	73
SRNet	72.3	76.2	79.4
TSRNet	**74.4**	**79.6**	**87.1**

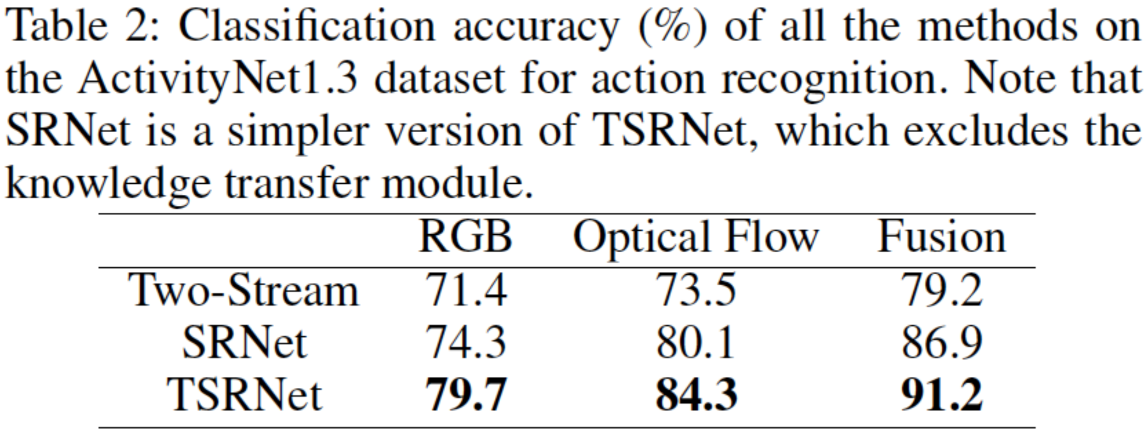


Table 2: Classification accuracy (%) of all the methods on the ActivityNet1.3 dataset for action recognition. Note that SRNet is a simpler version of TSRNet, which excludes the knowledge transfer module.

	RGB	Optical Flow	Fusion
Two-Stream	71.4	73.5	79.2
SRNet	74.3	80.1	86.9
TSRNet	**79.7**	**84.3**	**91.2**

The action recognition results. TSRNet performs good performance than the other based on weakly supervised learning scheme on THUMOS14 and ActivityNet1.3 datasets.

Accuracy

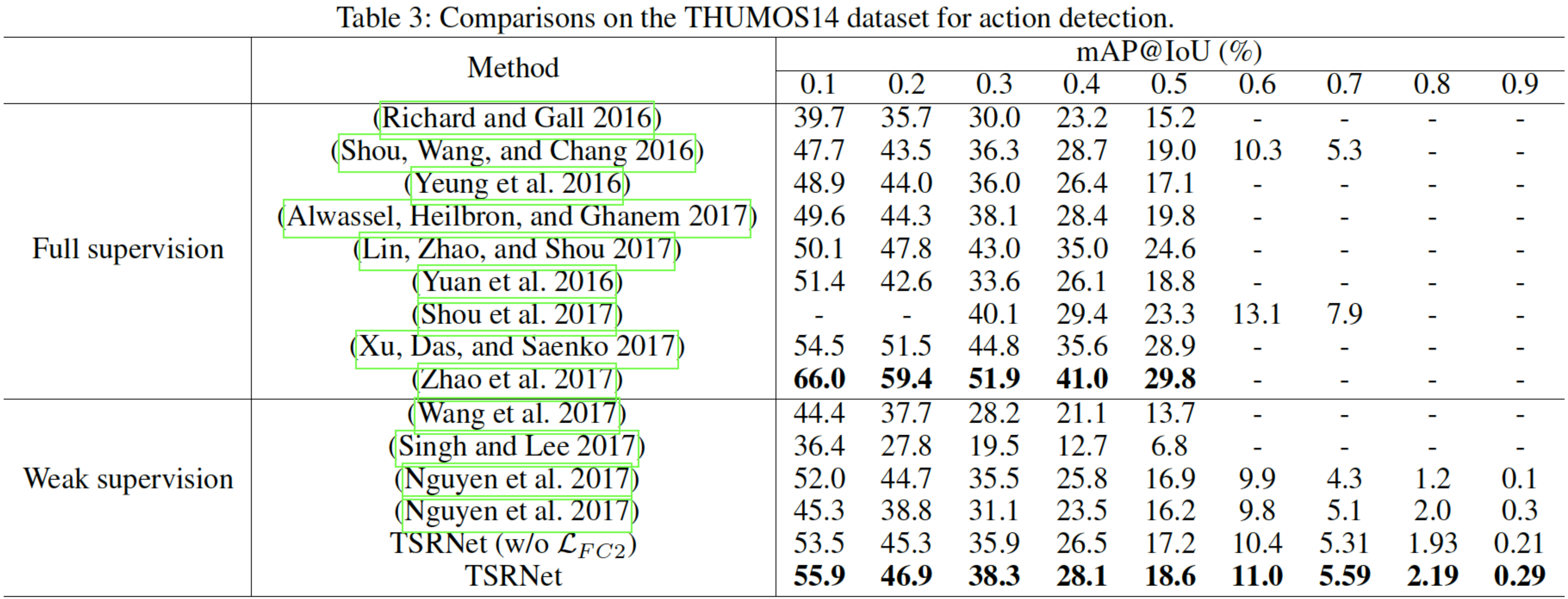


Table 3: Comparisons on the THUMOS14 dataset for action detection.

	Method	mAP@IoU (%)								
		0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9
Full supervision	(Richard and Gall 2016)	39.7	35.7	30.0	23.2	15.2	-	-	-	-
	(Shou, Wang, and Chang 2016)	47.7	43.5	36.3	28.7	19.0	10.3	5.3	-	-
	(Yeung et al. 2016)	48.9	44.0	36.0	26.4	17.1	-	-	-	-
	(Alwassel, Heilbron, and Ghanem 2017)	49.6	44.3	38.1	28.4	19.8	-	-	-	-
	(Lin, Zhao, and Shou 2017)	50.1	47.8	43.0	35.0	24.6	-	-	-	-
	(Yuan et al. 2016)	51.4	42.6	33.6	26.1	18.8	-	-	-	-
	(Shou et al. 2017)	-	-	40.1	29.4	23.3	13.1	7.9	-	-
	(Xu, Das, and Saenko 2017)	54.5	51.5	44.8	35.6	28.9	-	-	-	-
	(Zhao et al. 2017)	**66.0**	**59.4**	**51.9**	**41.0**	**29.8**	-	-	-	-
Weak supervision	(Wang et al. 2017)	44.4	37.7	28.2	21.1	13.7	-	-	-	-
	(Singh and Lee 2017)	36.4	27.8	19.5	12.7	6.8	-	-	-	-
	(Nguyen et al. 2017)	52.0	44.7	35.5	25.8	16.9	9.9	4.3	1.2	0.1
	(Nguyen et al. 2017)	45.3	38.8	31.1	23.5	16.2	9.8	5.1	2.0	0.3
	TSRNet (w/o $\mathcal{L}_{FC2}$)	53.5	45.3	35.9	26.5	17.2	10.4	5.31	1.93	0.21
	TSRNet	**55.9**	**46.9**	**38.3**	**28.1**	**18.6**	**11.0**	**5.59**	**2.19**	**0.29**

TSRNet can not only outperform other weakly supervised learning methods, it can also outperform some fully supervised learning methods for action detection.

Accuracy

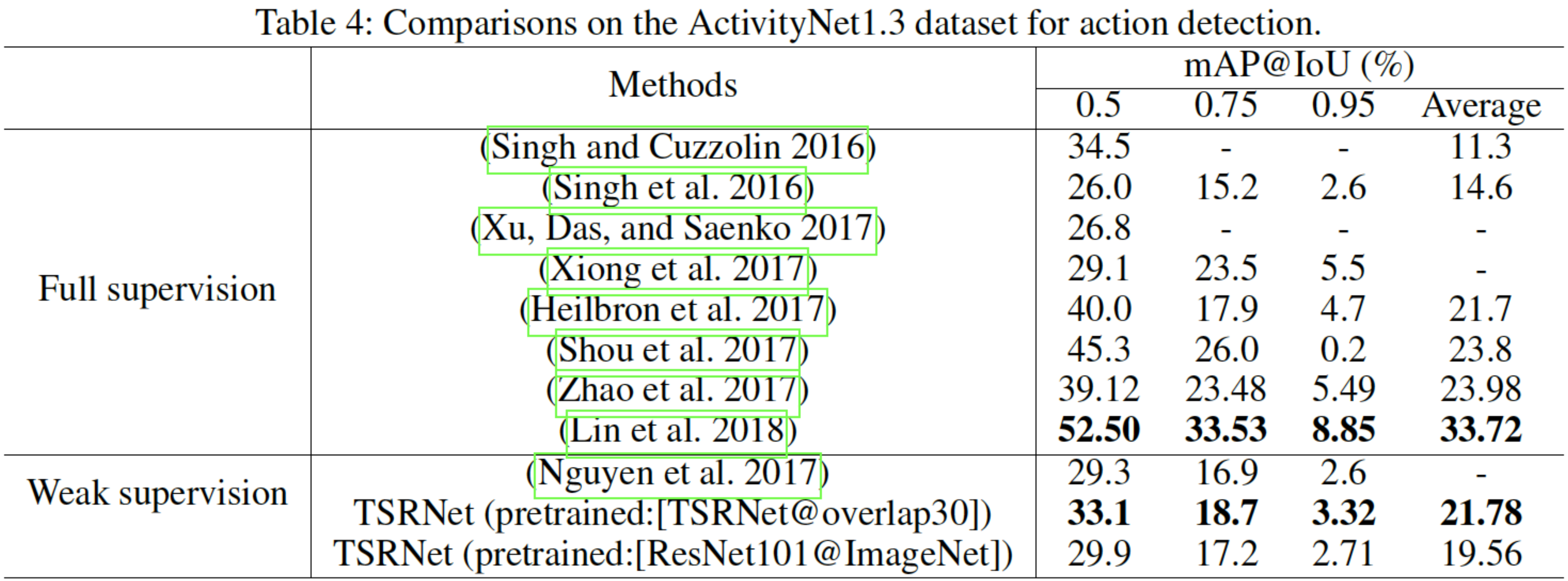


Table 4: Comparisons on the ActivityNet1.3 dataset for action detection.

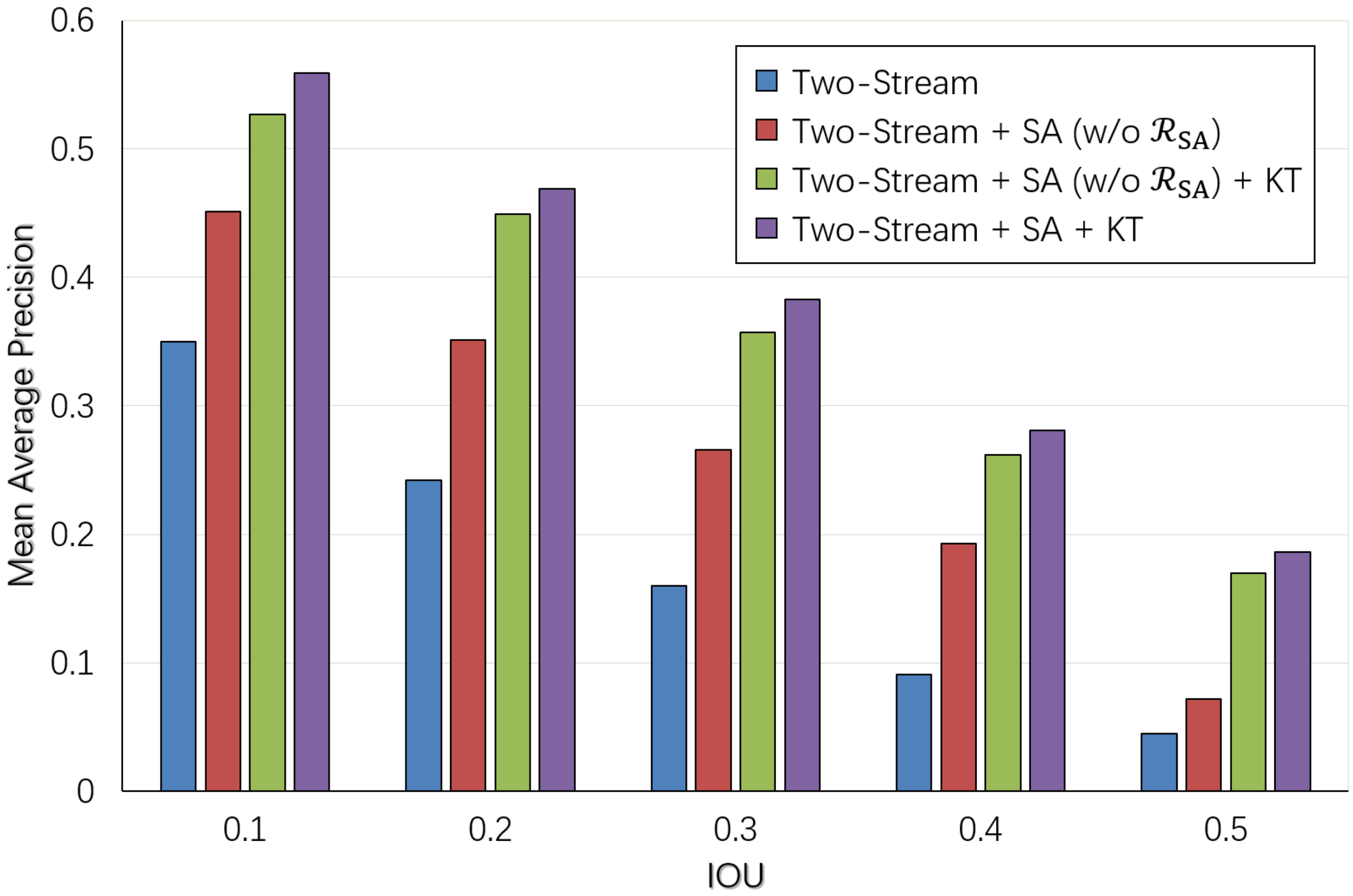
	Methods	mAP@IoU (%)			
		0.5	0.75	0.95	Average
Full supervision	(Singh and Cuzzolin 2016)	34.5	-	-	11.3
	(Singh et al. 2016)	26.0	15.2	2.6	14.6
	(Xu, Das, and Saenko 2017)	26.8	-	-	-
	(Xiong et al. 2017)	29.1	23.5	5.5	-
	(Heilbron et al. 2017)	40.0	17.9	4.7	21.7
	(Shou et al. 2017)	45.3	26.0	0.2	23.8
	(Zhao et al. 2017)	39.12	23.48	5.49	23.98
	(Lin et al. 2018)	**52.50**	**33.53**	**8.85**	**33.72**
Weak supervision	(Nguyen et al. 2017)	29.3	16.9	2.6	-
	TSRNet (pretrained:[TSRNet@overlap30])	**33.1**	**18.7**	**3.32**	**21.78**
	TSRNet (pretrained:[ResNet101@ImageNet])	29.9	17.2	2.71	19.56

Note that the ‘pretrained:[TSRNet@overlap30]' represents that we use the classes with overlapping labels found between the UCF101 and ActivityNet1.3 datasets to initialize the TSRNet and train it using the whole classes. The 'pretrained:[ResNet101@ImageNet]' represents that we use the ResNet101 pretrained on ImageNet dataset to initialize the TSRNet and then train it.

Outline

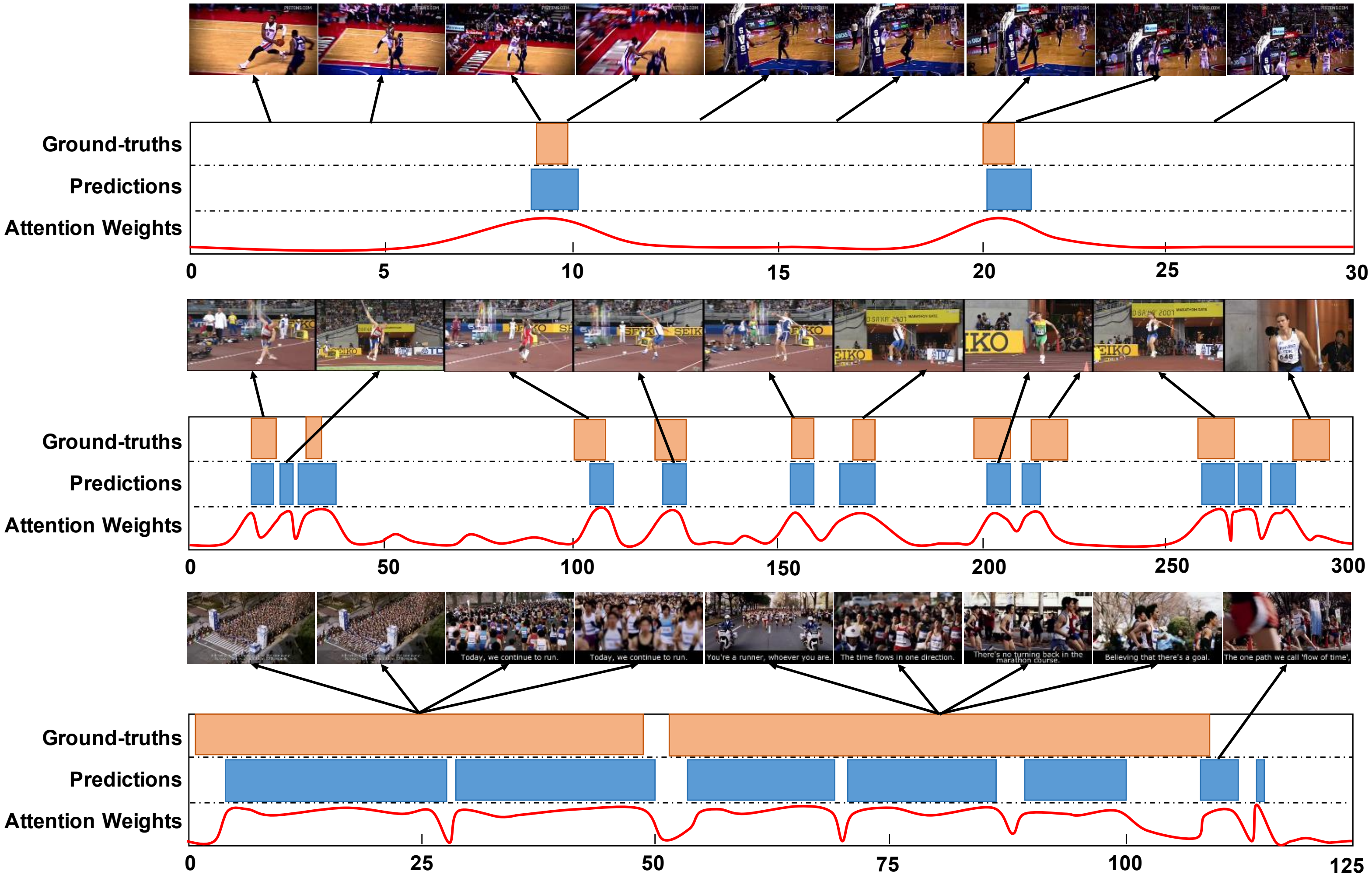
- 1. Introduction
  - Motivation
  - Overview
  - Related Work
- 2. Method (TSRNet)
  - Overview
  - Learning
  - Localization
- 3. Evaluation
  - Settings
  - Accuracy
  - Analysis

Analysis



The results of baselines and the full model among different IoUs. It shows that the self-attention with regularization loss and knowledge transfer contribute substantially to the model performance improvement.

Analysis



Qualitative results on THUMOS14 (top and middle) and ActivityNet1.3 (bottom).

Thank you!

Questions & Answers!