Embedded Multi-Person Pedestrian Tracking and Detection

MSCV19 Capstone Project, Internal(CMU)

Team Member: Yongxin Wang, Chunhui Liu
Advisor: Dr. Kris Kitani

02/15/2019

Introduction

- Motivation
  - Real-time multi-person pedestrain tracking
  - Visual analysis, automatic driving, robotics
- Problem
  - Detect multiple people
  - Keep tracking each of them
  - Deel with occulsion, large appearance changes
- Solution
  - Track by detection
  - Multiple template data association

Survey

- Single Object Tracking
  - High Performance Visual Tracking with Siamese Region Proposal Network
    - Bo Li, Junjie Yan, Wei Wu, Zheng Zhu, Xiaolin Hu, CVPR2018
  - Learning Multi-Domain Convolutional Neural Networks for Visual Tracking
    - Hyeonseob Nam, Bohyung Han CVPR2016
- Multi Object Tracking
  - Online Multi-Object Tracking with Dual Matching Attention Networks
    - Ji Zhu, Hua Yang, Nian Liu, Minyoung Kim, Wenjun Zhang, and Ming-Hsuan Yang, ECCV2018

Survey 1/3

High Performance Visual Tracking with Siamese Region Proposal Network
(CVPR 2018)

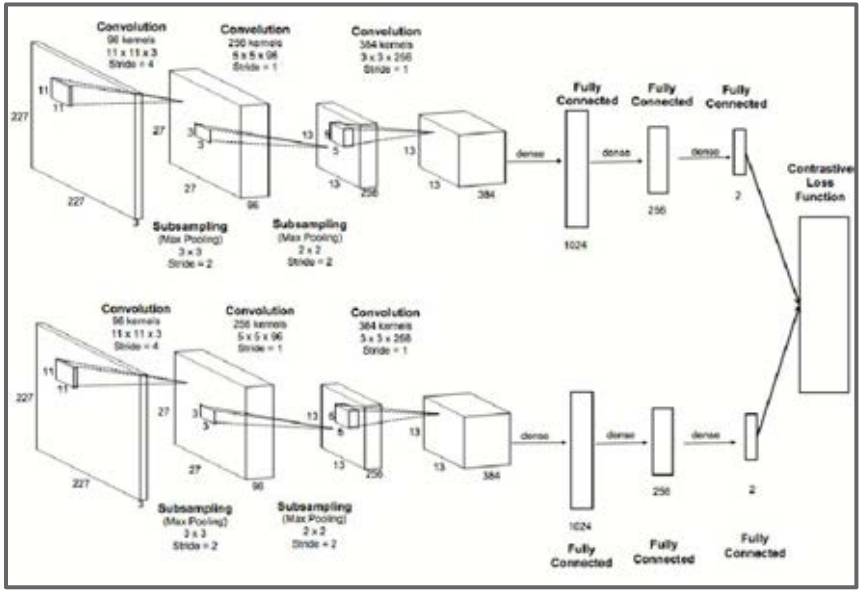
Bo Li[1,2], Junjie Yan[3], Wei Wu[1], Zheng Zhu[1,4,5], Xiaolin Hu[3]

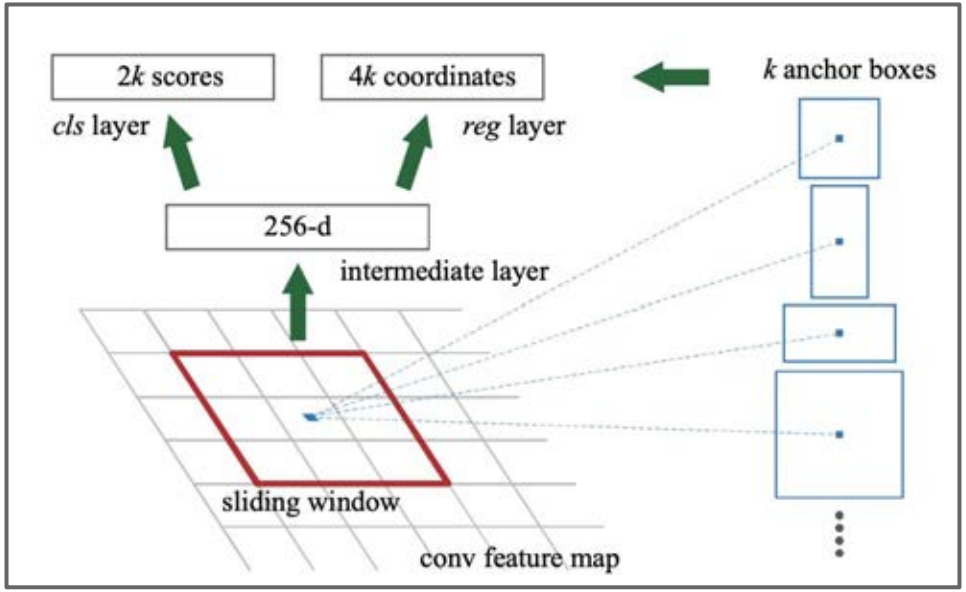
[1] SenseTime Group Limited [2] Beihang University [3] Tsinghua University
[4] Institute of Automation, Chinese Academy of Sciences
[5] University of Chinese Academy of Sciences

High Performance Visual Tracking with Siamese Region Proposal Network (SiameseRPN)

- The Siamese Twins - Chang and Eng, 1829 from Thailand

Siamese Network & Region Proposal Network





Siamese RPN Network

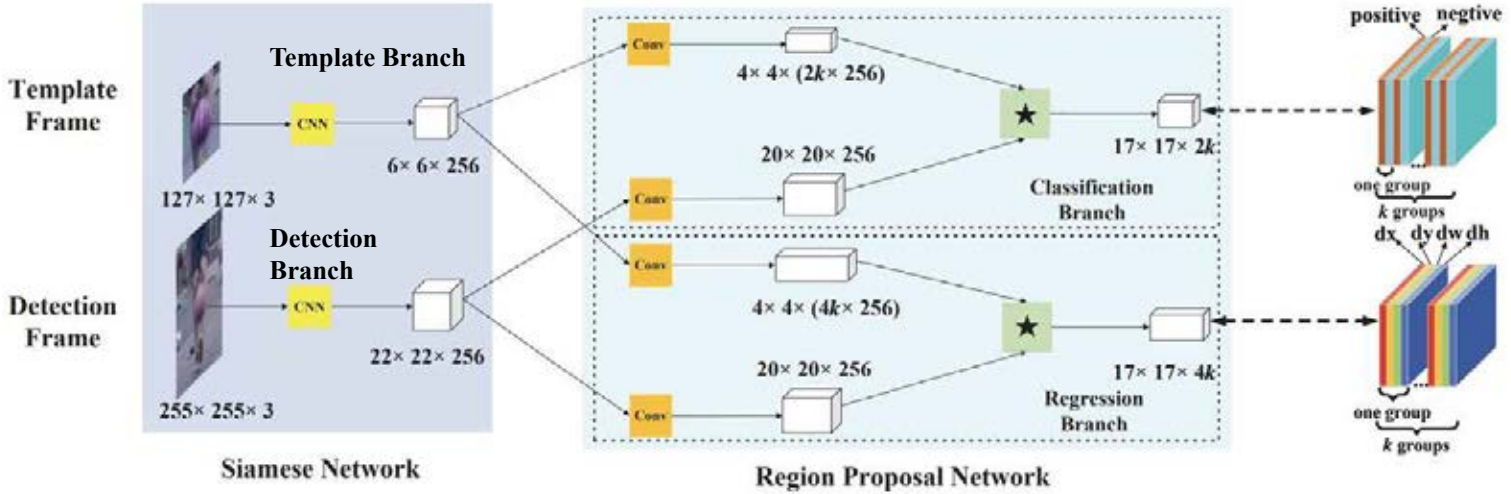


Figure 2: Main framework of Siamese-RPN: left side is Siamese subnetwork for feature extraction. Region proposal subnetwork lies in the middle, which has two branches, one for classification and the other for regression. Pair-wise correlation is adopted to obtain the output of two branches. Details of these two output feature maps are in the right side. In classification branch, the output feature map has $2k$ channels which corresponding to foreground and background of k anchors. In regression branch, the output feature map has $4k$ channels which corresponding to four coordinates used for proposal refinement of k anchors. In the figure, $\star$ denotes correlation operator.

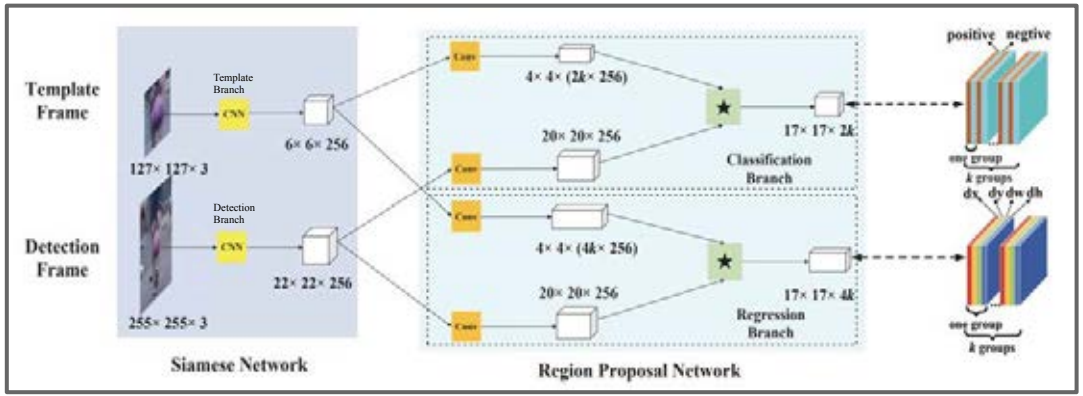
Training

- Data Set
  - Youtube-BB Dataset
  - ILSVRC Video Object Detection Dataset
- Training Scheme: End-to-End
- Losses:
  - Classification: Cross Entropy
  - Regression: Smooth L1 Loss

$$\delta[0] = \frac{T_x - A_x}{A_w}, \quad \delta[1] = \frac{T_y - A_y}{A_h}$$
$$\delta[2] = ln\frac{T_w}{A_w}, \quad \delta[3] = ln\frac{T_h}{A_h}$$

$$smooth_{L_1}(x, \sigma) = \begin{cases} 0.5\sigma^2 x^2, & |x| < \frac{1}{\sigma^2} \\ |x| - \frac{1}{2\sigma^2}, & |x| \geq \frac{1}{\sigma^2} \end{cases}$$

$$loss = L_{cls} + \lambda L_{reg}$$



Inference

- Template Selection
  - 1st frame template is used and fixed for the rest
- Proposal Selection
  - Discard proposals too far away
  - Re-Rank by Cosine window + scale penalty

$$penalty = e^{k*max(\frac{r}{r'}, \frac{r'}{r})*max(\frac{s}{s'}, \frac{s'}{s})}$$

$$(w+p) \times (h+p) = s^2$$

$$p = \frac{w+h}{2}$$

- Running Speed: 160 FPS

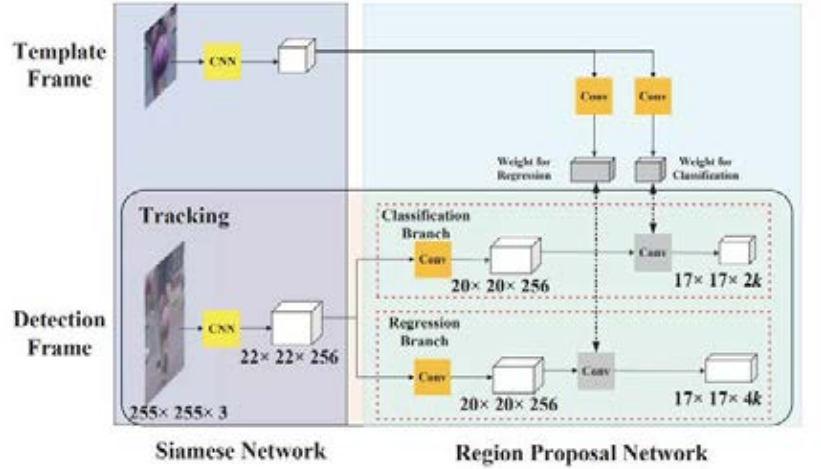


Figure 3: Tracking as one-shot detection: the template branch predicts the weights(in gray) for kernels of region proposal subnetwork on detection branch using the first frame. Then the template branch is pruned and only the detection branch is retained. So the framework is modified to a local detection network.

Results: Visual Object Tracking (VOT) Challenge

Tracker	EAO	Accuracy	Failure	EFO
DeepSRDCF	*0.3181*	*0.56*	*1.0*	0.38
EBT	*0.313*	0.45	*1.02*	1.76
SRDCF	0.2877	0.55	1.18	1.99
LDP	0.2785	0.49	1.3	4.36
sPST	0.2767	0.54	1.42	1.01
SC-EBT	0.2548	0.54	1.72	0.8
NSAMF	0.2536	0.53	1.29	5.47
Struck	0.2458	0.46	1.5	2.44
RAJSSC	0.242	*0.57*	1.75	2.12
S3Tracker	0.2403	0.52	1.67	*14.27*
SiamFC-3s	0.2915	0.54	1.42	*8.68*
SiamFC-5s	0.275	0.53	1.45	7.84
SiamRPN	***0.358***	***0.58***	***0.93***	***23.0***

Table 1: Details about the state-of-the-art trackers in VOT2015. ***Red***, *blue* and *green*, represent *1st*, *2nd* and *3rd* respectively.

Tracker	EAO	Accuracy	Failure	EFO
C-COT	*0.331*	0.53	***0.85***	0.507
ECO-HC	*0.322*	0.53	*1.08*	*15.13*
Staple	0.2952	*0.54*	1.35	*11.14*
EBT	0.2913	0.47	*0.9*	3.011
MDNet_N	0.257	*0.54*	1.2	0.534
SiamRN	0.2766	*0.55*	1.37	5.44
SiamAN	0.2352	0.53	1.65	9.21
SiamRPN	***0.3441***	***0.56***	*1.08*	***23.3***

Table 2: Detail information about several published state-of-the-art trackers' performances in VOT2016.

Results: Object Tracking Benchmark (OTB100)

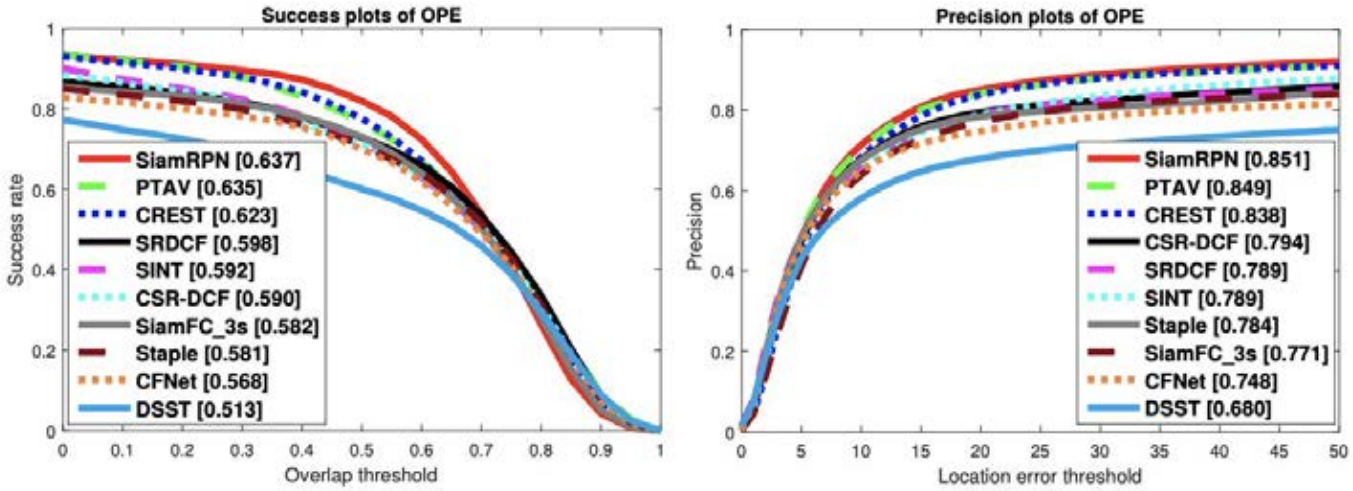


Figure 9: Success plot and precision plot of OTB2015

Survey 2/3

Learning Multi-Domain Convolutional Neural Networks for Visual Tracking (CVPR2016)

Hyeonseob Nam, Bohyung Han
Dept. of Computer Science and Engineering, POSTECH, Korea

Learning Multi-Domain Convolutional Neural Networks for Visual Tracking (MD-Net)

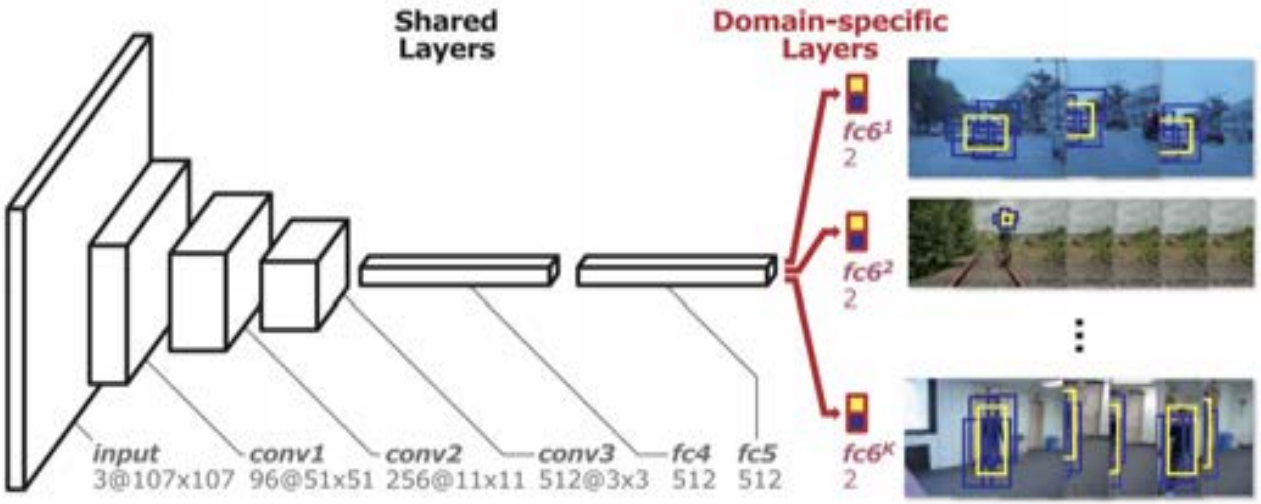


Figure 1: The architecture of our Multi-Domain Network, which consists of shared layers and K branches of domain-specific layers. Yellow and blue bounding boxes denote the positive and negative samples in each domain, respectively.

Training

- Dataset
  - VOT2014, OTB
- Training Scheme: Offline Pretraining + Online training (at inference)
- Losses:
  - Cross entropy loss

Offline Pre-Training

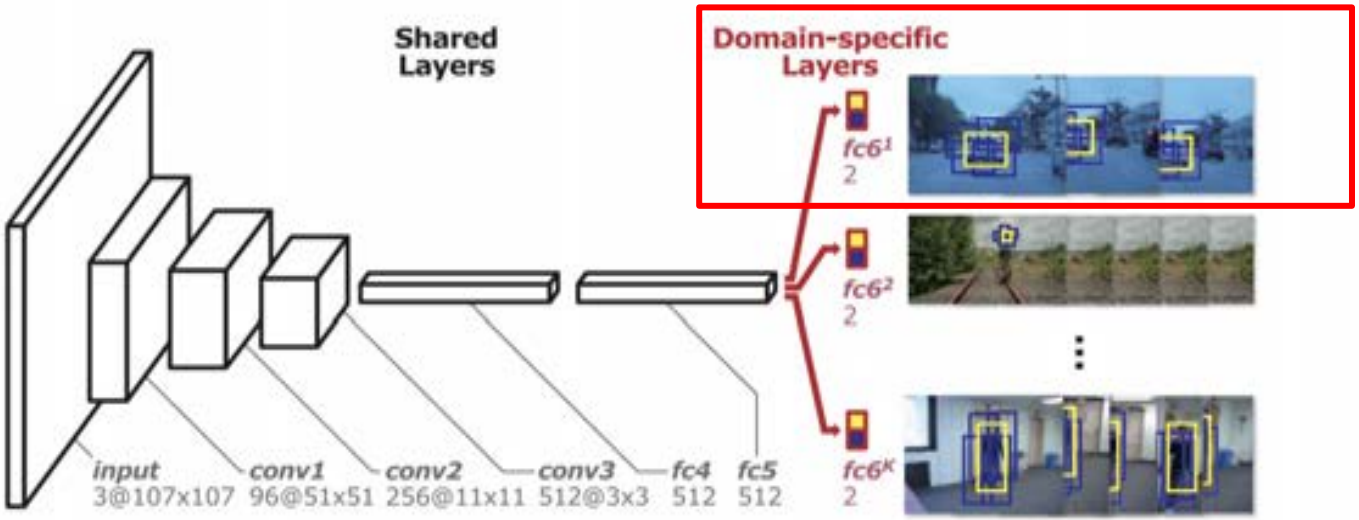


Figure 1: The architecture of our Multi-Domain Network, which consists of shared layers and K branches of domain-specific layers. Yellow and blue bounding boxes denote the positive and negative samples in each domain, respectively.

Offline Pre-Training

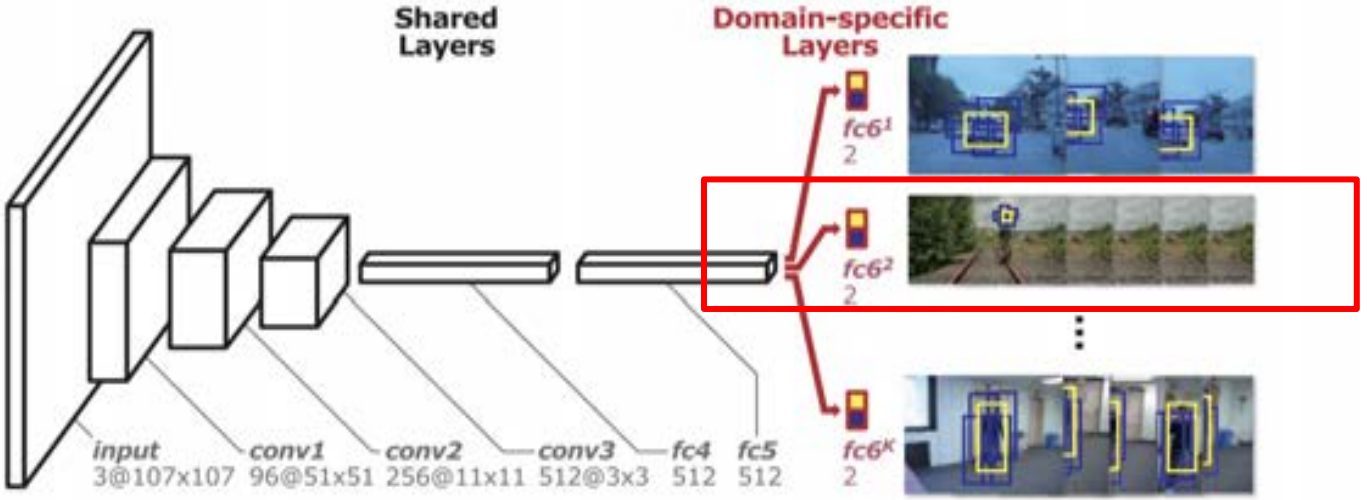


Figure 1: The architecture of our Multi-Domain Network, which consists of shared layers and K branches of domain-specific layers. Yellow and blue bounding boxes denote the positive and negative samples in each domain, respectively.

Offline Pre-Training

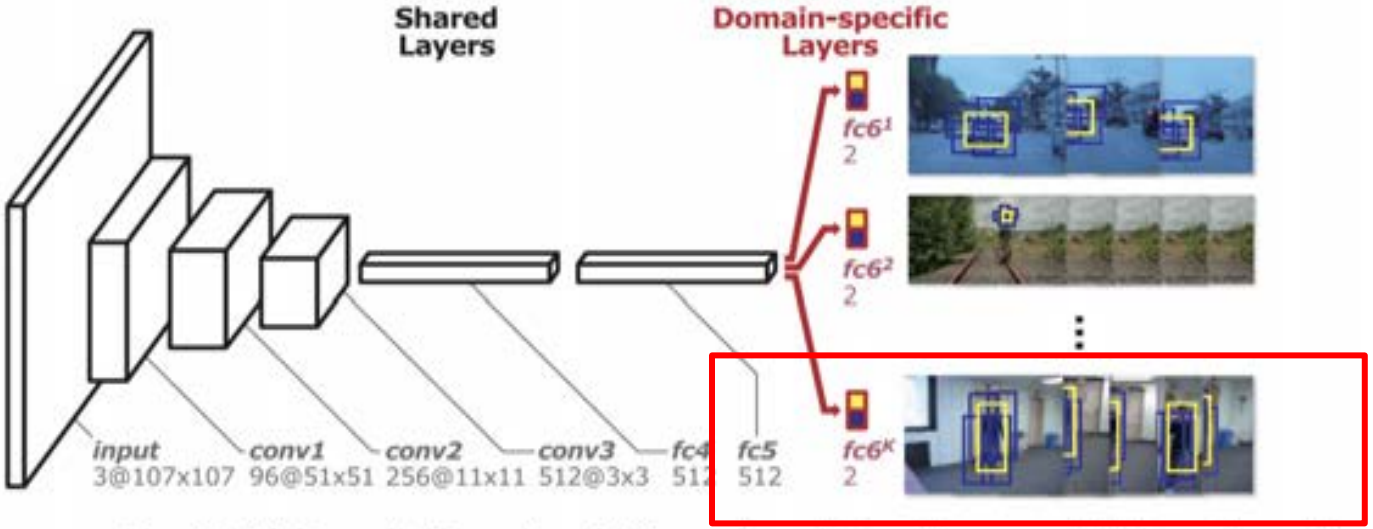


Figure 1: The architecture of our Multi-Domain Network, which consists of shared layers and K branches of domain-specific layers. Yellow and blue bounding boxes denote the positive and negative samples in each domain, respectively.

Inference - Online Training

- Discard all offline fc6's
- Reinitialize a new fc6 for current video
- Weight update (fc4-fc6) only when:
  - $\mathbf{x}^*$ drops below 0.5
  - every 10 frames
- Bounding Box Regression
  - Only trained on first test frame
  - Run on other frames
- Hard Mini-batch mining
  - M^+ positive examples
  - Top M_h^- hard negative examples drawn from M^- negative examples ($M^- >> M_h^-$)
- Running Speed: ~50 FPS

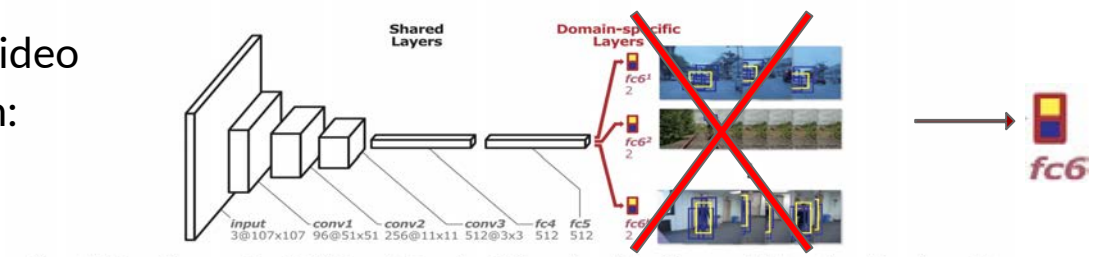
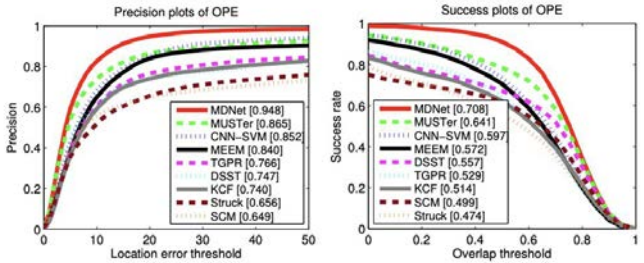
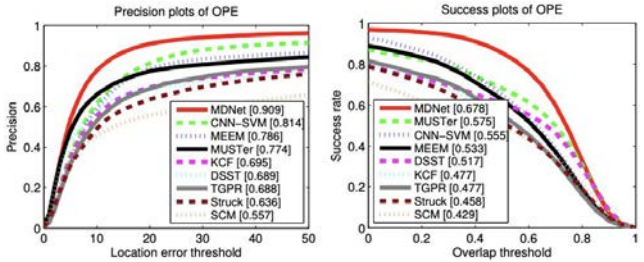


Figure 1: The architecture of our Multi-Domain Network, which consists of shared layers and K branches of domain-specific layers. Yellow and blue bounding boxes denote the positive and negative samples in each domain, respectively.

Results - OTB50/100



(a) OTB50 result



(b) OTB100 result

Figure 3: Precision and success plots on OTB50 [46] and OTB100 [45]. The numbers in the legend indicate the representative precision at 20 pixels for precision plots, and the average area-under-curve scores for success plots.

Results - VOT 2014

Tracker	Accuracy		Robustness		Combined
	Score	Rank	Score	Rank	Rank
MUSTer	0.58	4.50	0.99	5.67	5.09
MEEM	0.48	7.17	0.71	5.50	6.34
DSST	0.60	4.03	0.68	5.17	4.60
SAMF	0.60	3.97	0.77	5.58	4.78
KCF	0.61	3.82	0.79	5.67	4.75
DGT	0.53	4.49	0.55	3.58	4.04
PLT_14	0.53	5.58	0.14	2.75	4.17
MDNet	0.63	2.50	0.16	2.08	2.29

(a) Baseline result

Tracker	Accuracy		Robustness		Combined
	Score	Rank	Score	Rank	Rank
MUSTer	0.55	4.67	0.94	5.53	5.10
MEEM	0.48	7.25	0.74	5.76	6.51
DSST	0.58	4.00	0.76	5.10	4.55
SAMF	0.57	3.72	0.81	4.94	4.33
KCF	0.58	3.92	0.87	4.99	4.46
DGT	0.54	3.58	0.67	4.17	3.88
PLT_14	0.51	5.43	0.16	2.08	3.76
MDNet	0.60	3.31	0.30	3.58	3.45

(b) Region_noise result

Table 1: The average scores and ranks of accuracy and robustness on the two experiments in VOT2014 [26]. The first and second best scores are highlighted in red and blue colors, respectively.

Results - VOT 2014

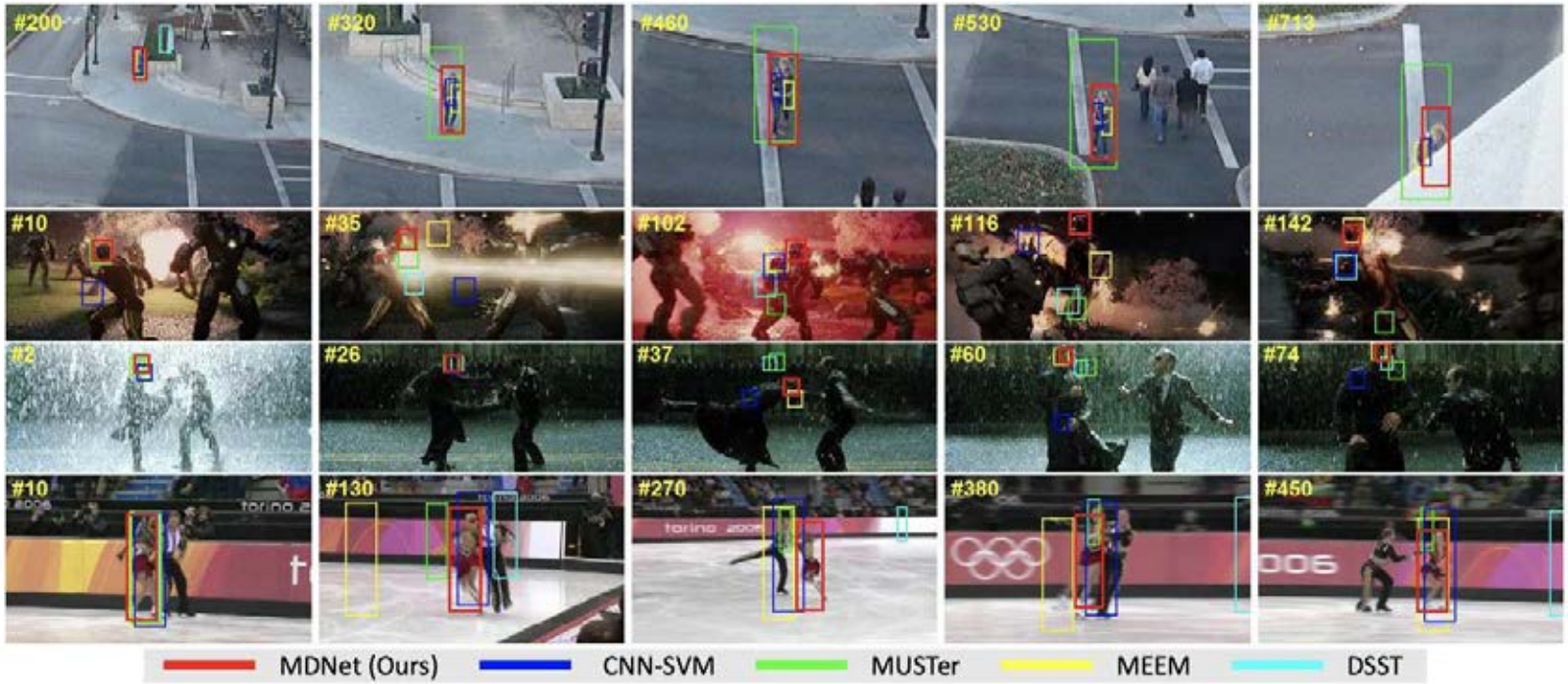


Figure 6: Qualitative results of the proposed method on some challenging sequences (*Bolt2*, *Diving*, *Freeman4*, *Human5*, *Ironman*, *Matrix* and *Skating2-1*).

Real Time MD-Net *(I. Jung, J. Son, M. Baek, and B. Han. Real-time mdnet. In ECCV, 2018.)*

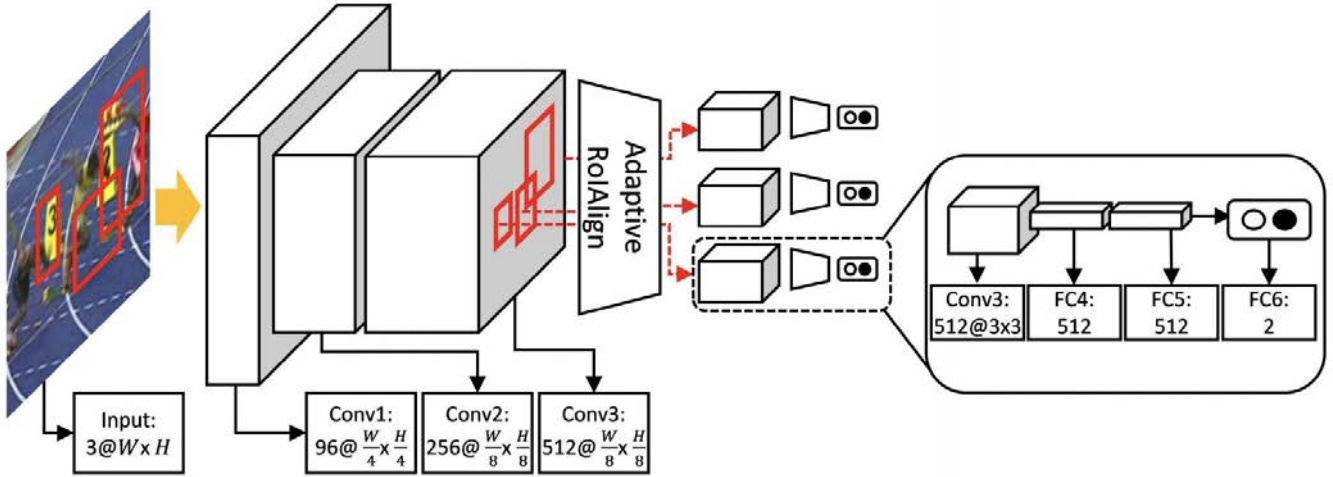


Fig. 1. Network architecture of the proposed tracking algorithm. The network is composed of three convolutional layers for extracting a shared feature map, adaptive RoIAlign layer for extracting a specific feature using regions of interest (RoIs), and three fully connected layers for binary classification. The number of channels and the size of each feature map are shown with the name of each layer.

Survey 3/3

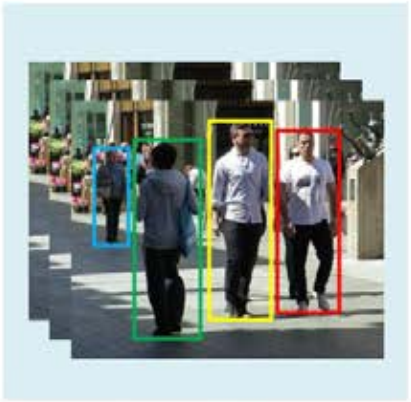
Online Multi-Object Tracking with Dual Matching Attention Networks
(ECCV2018)

Ji Zhu[1,2], Hua Yang[1], Nian Liu[3], Minyoung Kim[4], Wenjun Zhang[1], and Ming-Hsuan Yang[5,6]

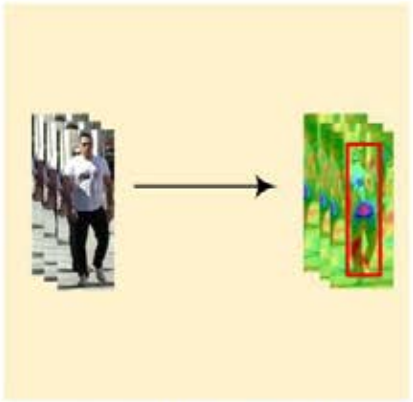
[1]Shanghai Jiao Tong University [2]Visbody Inc
[3]Northwestern Polytechnical University [4]Massachusetts Institute of Technology
[5]University of California, Merced [6]Google Inc

Online Multi-Object Tracking with Dual Matching Attention Networks

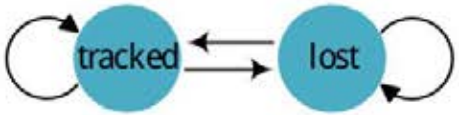
- Multi-Object Tracking Formulation
  - Detection + tracking + data association
  - 1. For new detection box, apply a single object tracker
  - 2. For unreliable trackers, associate this tracker with a new detection result

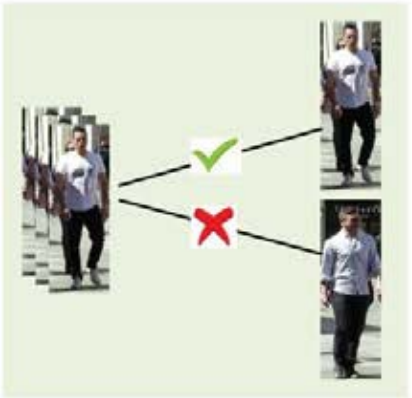


Detection



Single object tracking

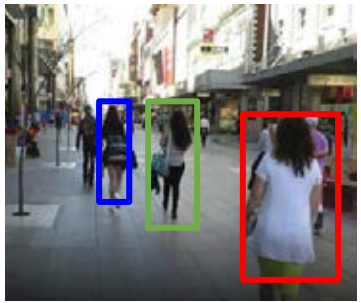




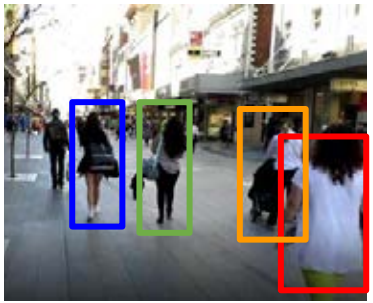
Data association

Framework

Detection | Tracking | Data Association



Detection Frame 0

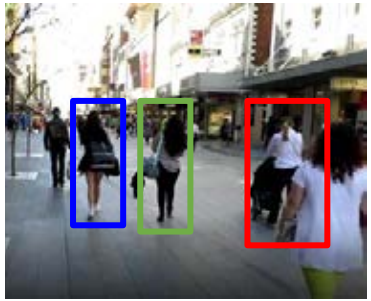


Detection Frame 0+t

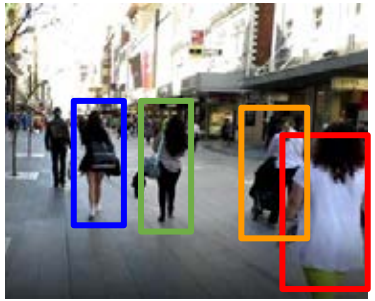
Tracker 1 — High Score

Tracker 2 — High Score

Tracker 3 — Low Score



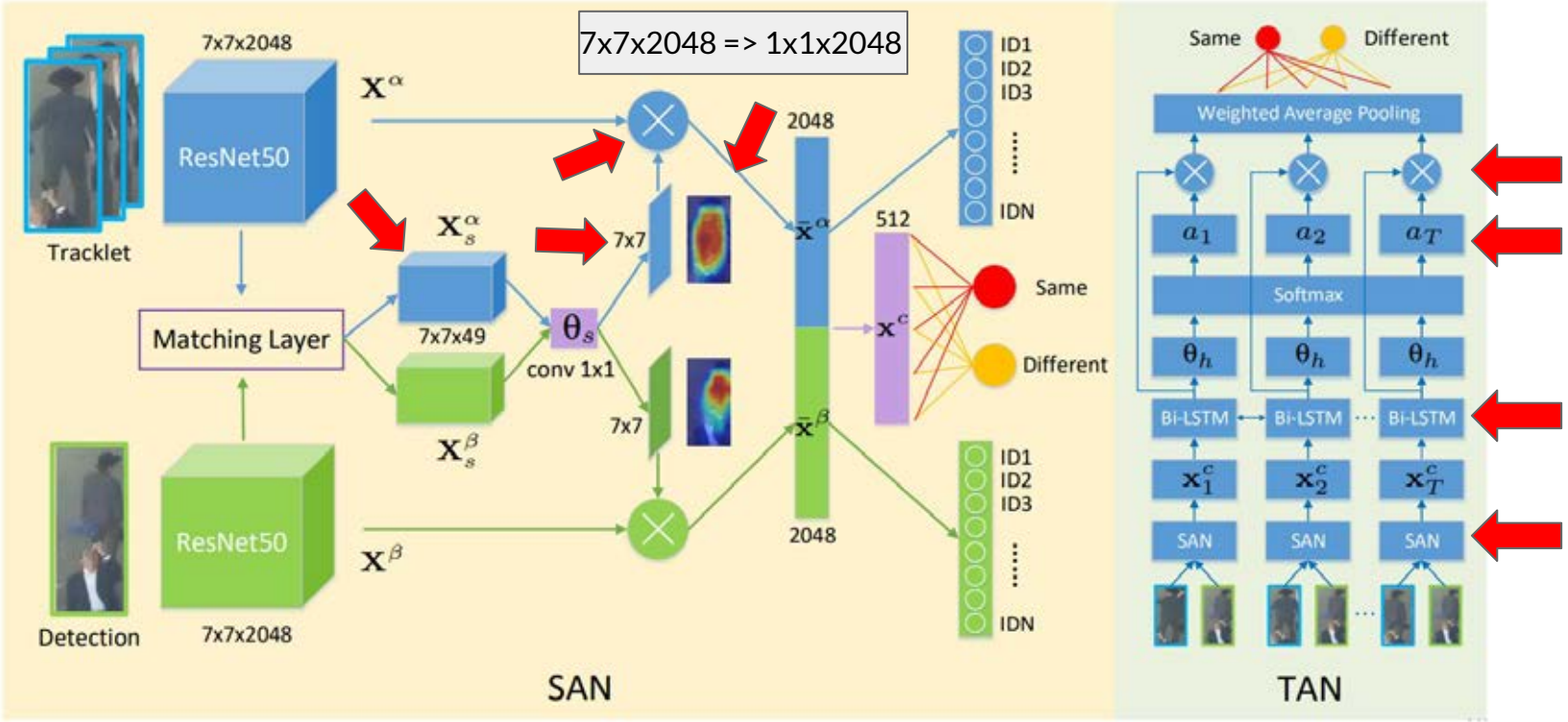
Tracking Result Frame 0...t-1



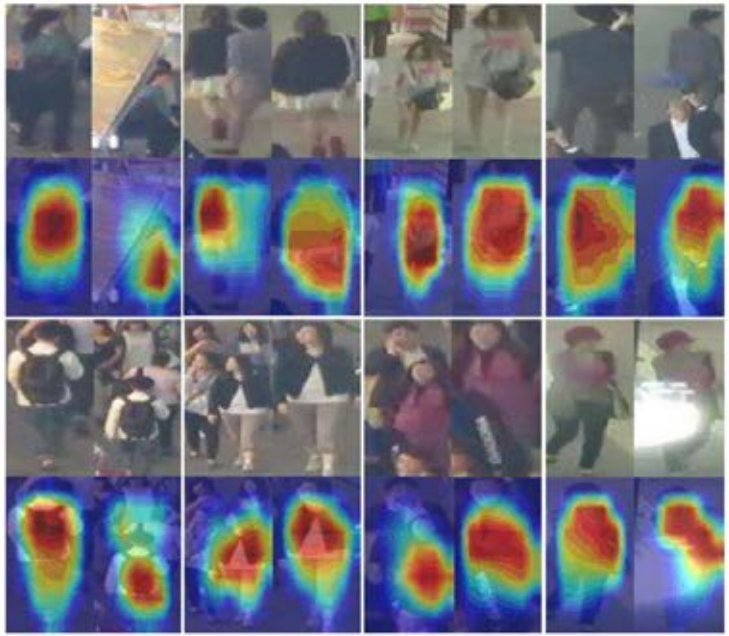
Detection @ Frame 0+t

Association Network

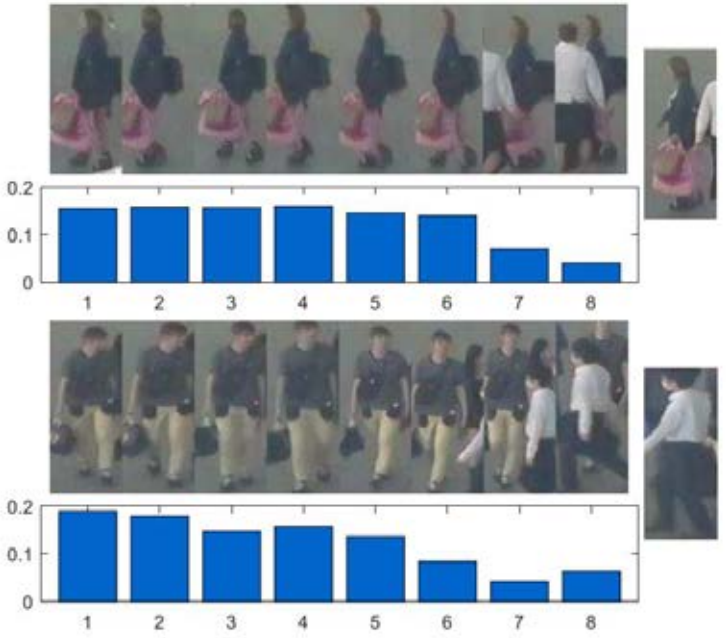
Spatial-Temporal Attention Association Network



Results: Attention Results



(a) Spatial attention maps



(b) Temporal attention values

Results: Multi Object Tracking Results

Table 1. Tracking performance on the MOT16 dataset.

Mode	Method	MOTA ↑	MOTP ↑	IDF ↑	IDP ↑	IDR ↑	MT ↑	ML ↓	FP ↓	FN ↓	IDS ↓	Frag ↓	AR ↓
Online	OVBT [3]	38.4	75.4	37.8	55.4	28.7	7.5%	47.3%	11,517	99,463	1,321	2,140	49.8
	EAMTT [43]	38.8	75.1	42.4	65.2	31.5	7.9%	49.1%	8,114	102,452	965	1,657	37.4
	oICF [22]	43.2	74.3	49.3	73.3	37.2	11.3%	48.5%	6,651	96,515	**381**	**1,404**	33.3
	CDA_DDAL [2]	43.9	74.7	45.1	66.5	34.1	10.7%	44.4%	6,450	95,175	676	1,795	31.8
	STAM [10]	46.0	74.9	50.0	71.5	38.5	14.6%	43.6%	6,895	91,117	473	1,422	29.6
	AMIR [42]	**47.2**	**75.8**	46.3	68.9	34.8	14.0%	**41.6%**	**2,681**	92,856	774	1,675	21.8
	Ours	46.1	73.8	**54.8**	**77.2**	**42.5**	**17.4%**	42.7%	7,909	**89,874**	532	1,616	**19.3**
Offline	QuadMOT [45]	44.1	76.4	38.3	56.3	29.0	14.6%	44.9%	6,388	94,775	745	1,096	31.9
	EDMT [7]	45.3	75.9	47.9	65.3	37.8	17.0%	39.9%	11,122	87,890	639	946	20.3
	MHT_DAM [23]	45.8	76.3	46.1	66.3	35.3	16.2%	43.2%	6,412	91,758	590	781	23.7
	JMC [47]	46.3	75.7	46.3	66.3	35.6	15.5%	**39.7%**	6,373	90,914	657	1,114	21.1
	NOMT [9]	46.4	76.6	**53.3**	**73.2**	**41.9**	18.3%	41.4%	9,753	87,565	**359**	**504**	16.3
	MCjoint [21]	47.1	76.3	52.3	73.9	40.4	**20.4%**	46.9%	6,703	89,368	370	598	18.6
	NLLMPa [29]	47.6	78.5	47.3	67.2	36.5	17.0%	40.4%	**5,844**	89,093	629	768	16.8
	LMP [48]	**48.8**	**79.0**	51.3	71.1	40.1	18.2%	40.1%	6,654	**86,245**	481	595	**14.8**

Conclusion

	Real-Time	Template Update	Uniform Aritecture	Multi-Person
MD-Net	No	Yes	Yes	No
SiamRPN	Yes	No	Yes	No
DMA Net	No	Yes	No	Yes
Ours	Yes	Yes	Yes	Yes