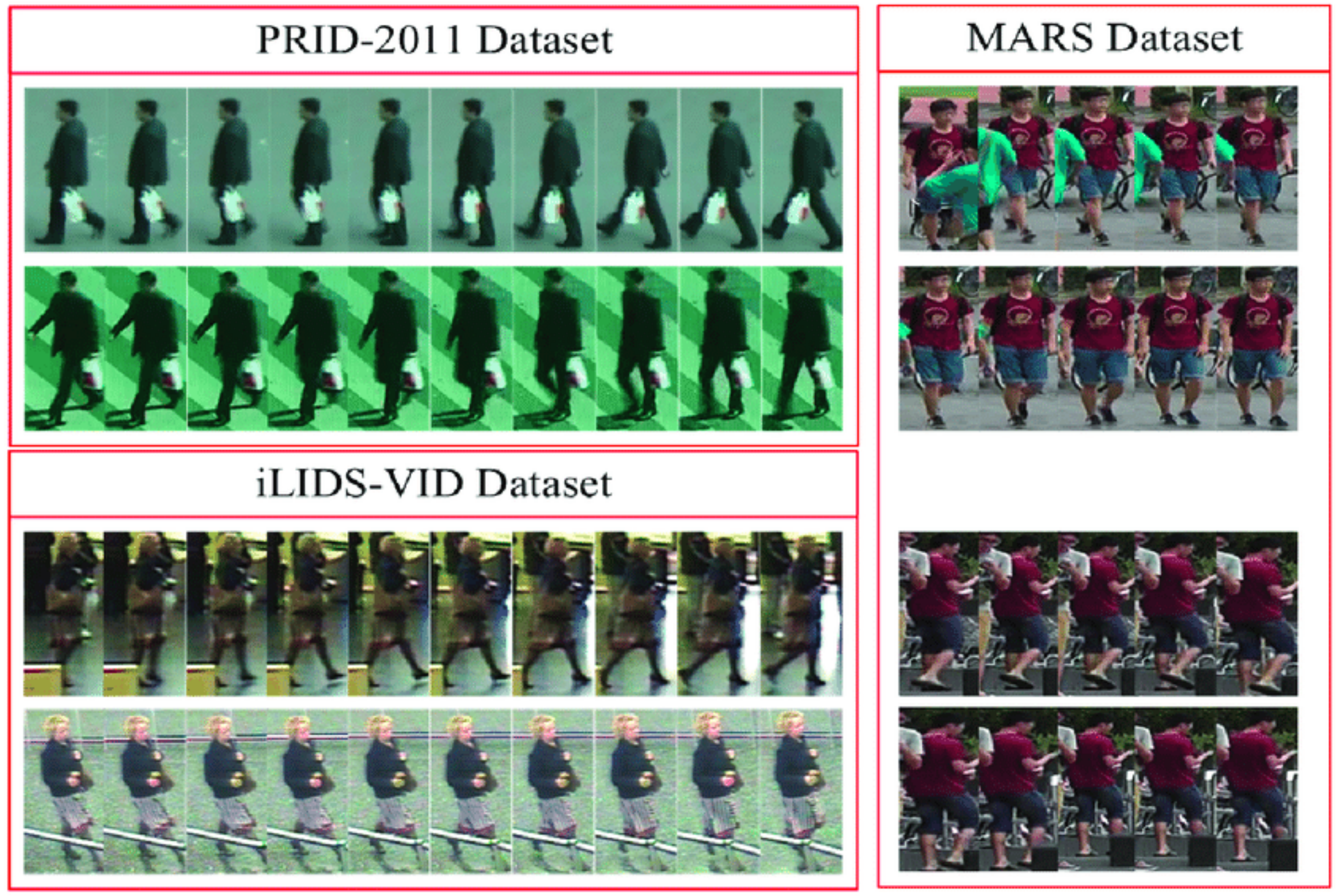
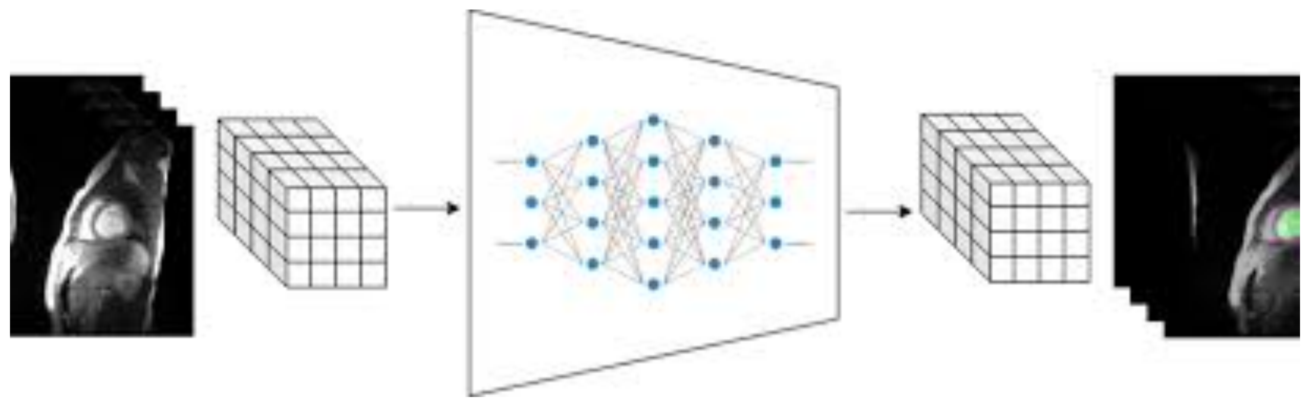
SSN3D: Self-Separated Network to Align Parts for 3D Convolution in Video Person Re-Identification

Xiaoke JIANG <jiangxiaoke@sensetime.com>
Joint work with Yu Qiao, Junjie Yan, Qichen Li, Wanrong Zheng, Dapeng Chen

Video Person Re-identification





3D Convolutional Network

For each individual, we have several video clips describing him.

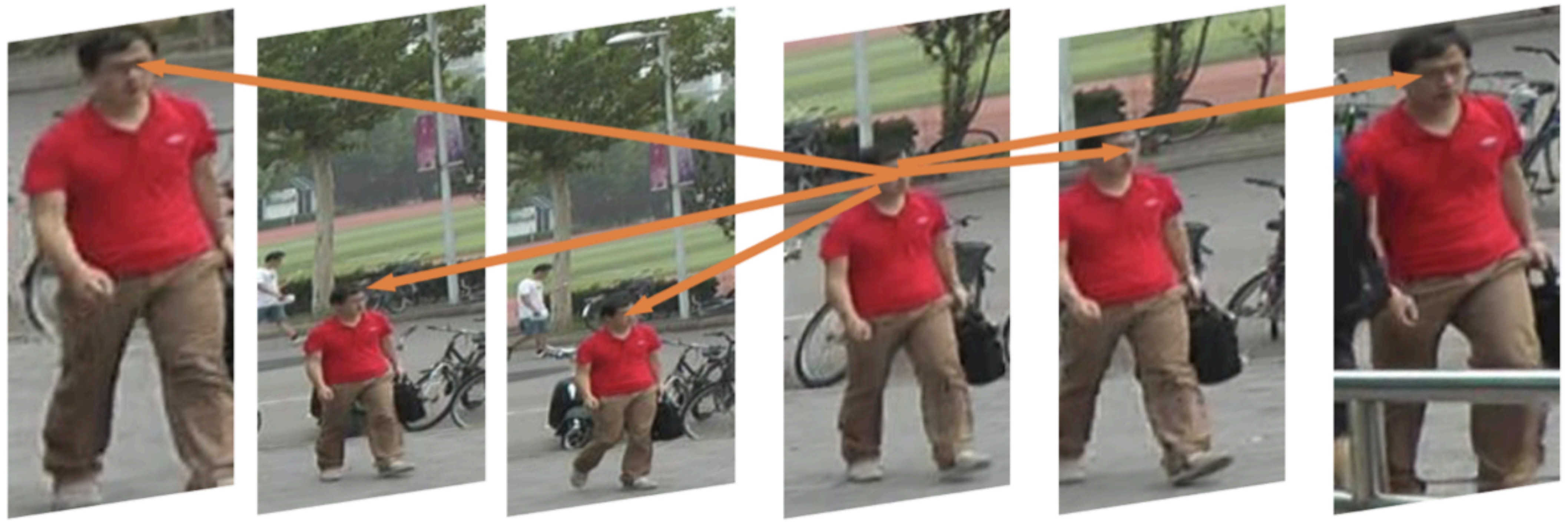
Each video clip consists of several frames of bounding boxes of a person.

Our task is that given a new video clips(query), we want to find the same person in existing gallery.

What is special?

We will have to involve temporal modeling to describe the temporal information in each video clip.

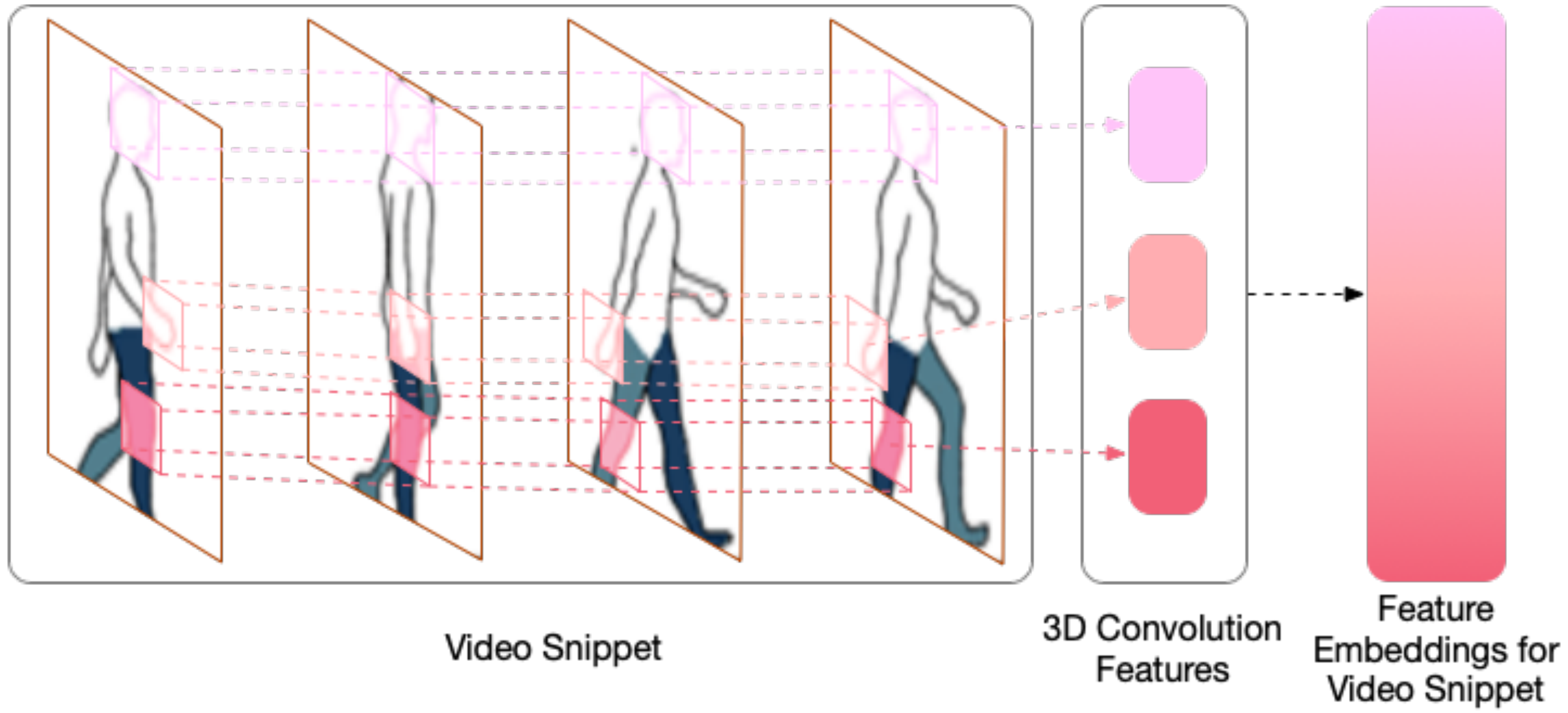
Existing Challenge in 3D CNN



Misalignment Problem in Temporal Dimension

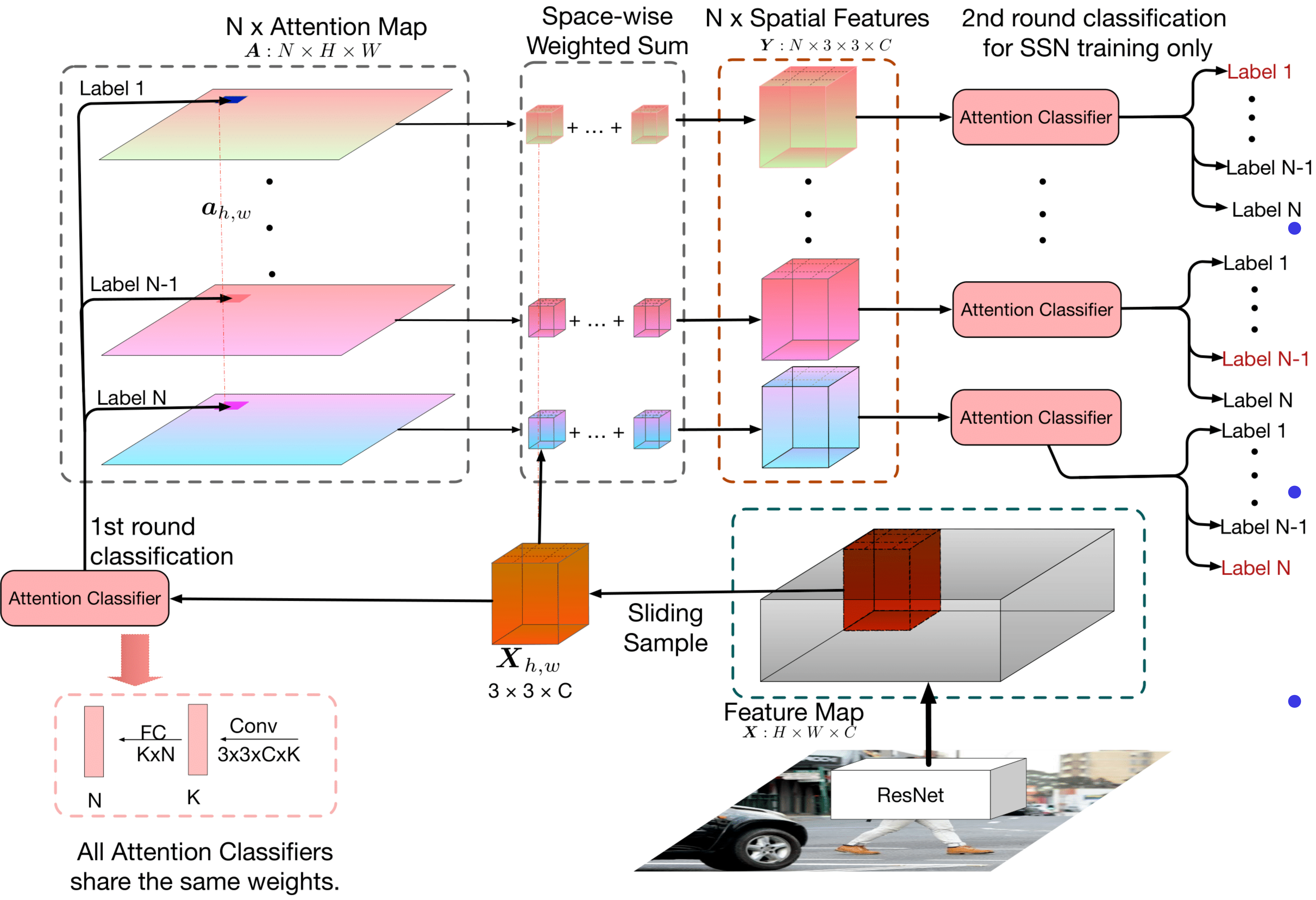
- Imperfect bounding box detection
- Constantly changing posture of a pedestrian
- The appearance of occlusion

Introduction — What We Do



- We proposed a new alignment mechanism, namely region proposal networks, to address the challenges in the 3D convolutional network.
- Our experiment shows that the fixed-attention training scheme has an extraordinary learning ability that it can even learn features from the data generated by random distribution.
- When we apply the model to real-world tasks like video-based ReID. We have achieved superior performance compared with state-of-the-art methods.

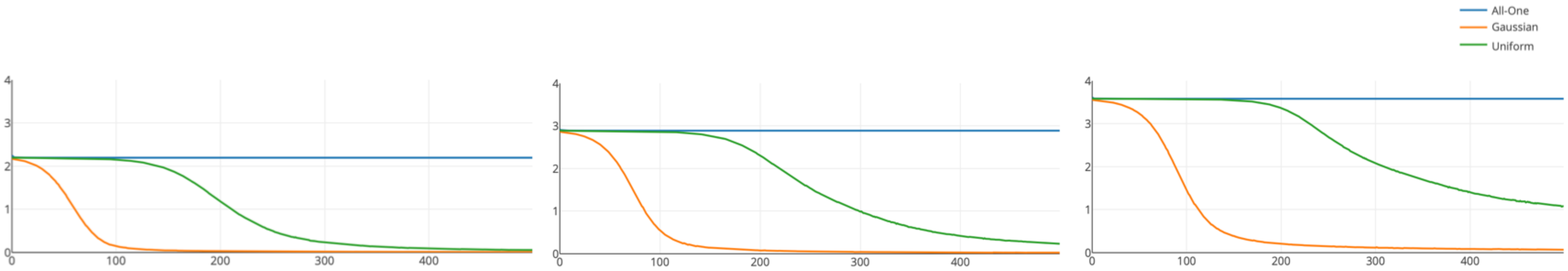
SSN: Self-Separated Network



- The weight-sharing attention classifiers secure the consistency between spatial and temporal features, and guarantees the effectiveness of unsupervised learning.
- The attention maps describe different attention with regard of different region of interested(head, hand, or feet).
- Semi-supervised learning is easy to be introduced into our model.

Experimental Results on RAN

Strong Learning Ability



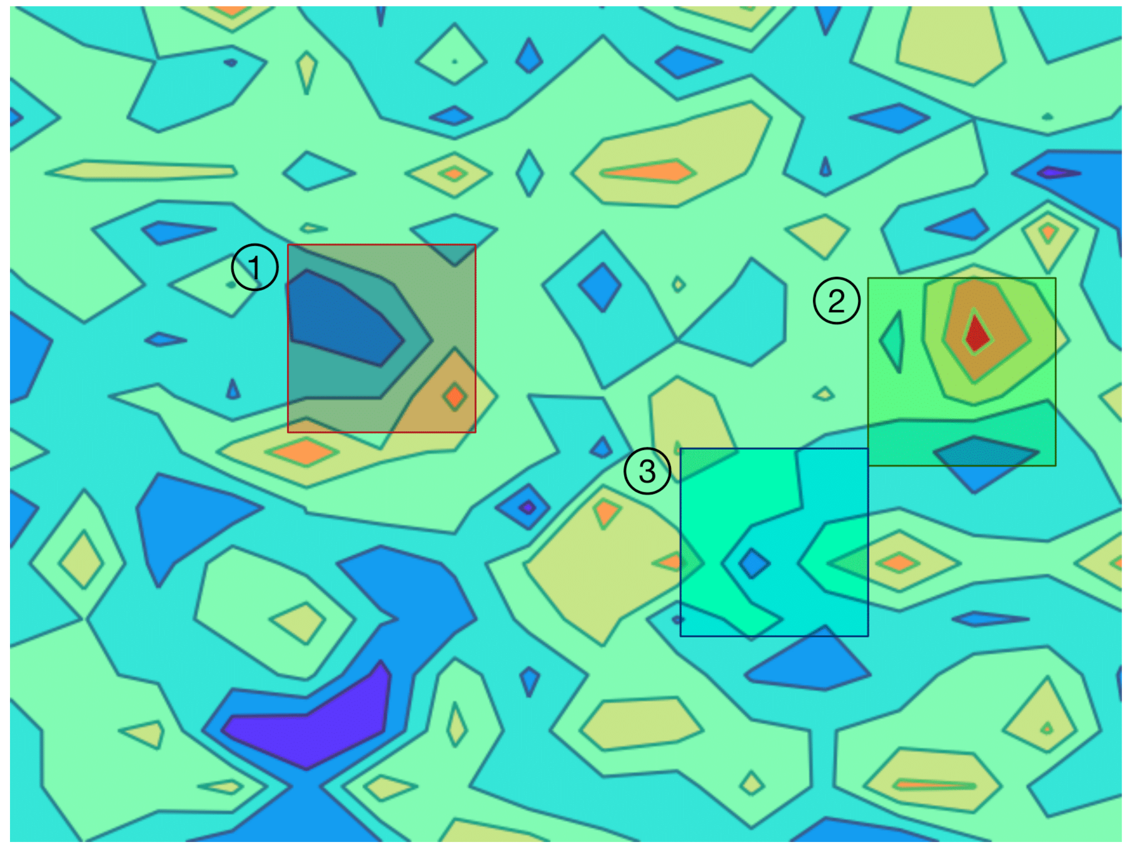
(a) Cross-Entropy Loss on 9-Label RAN

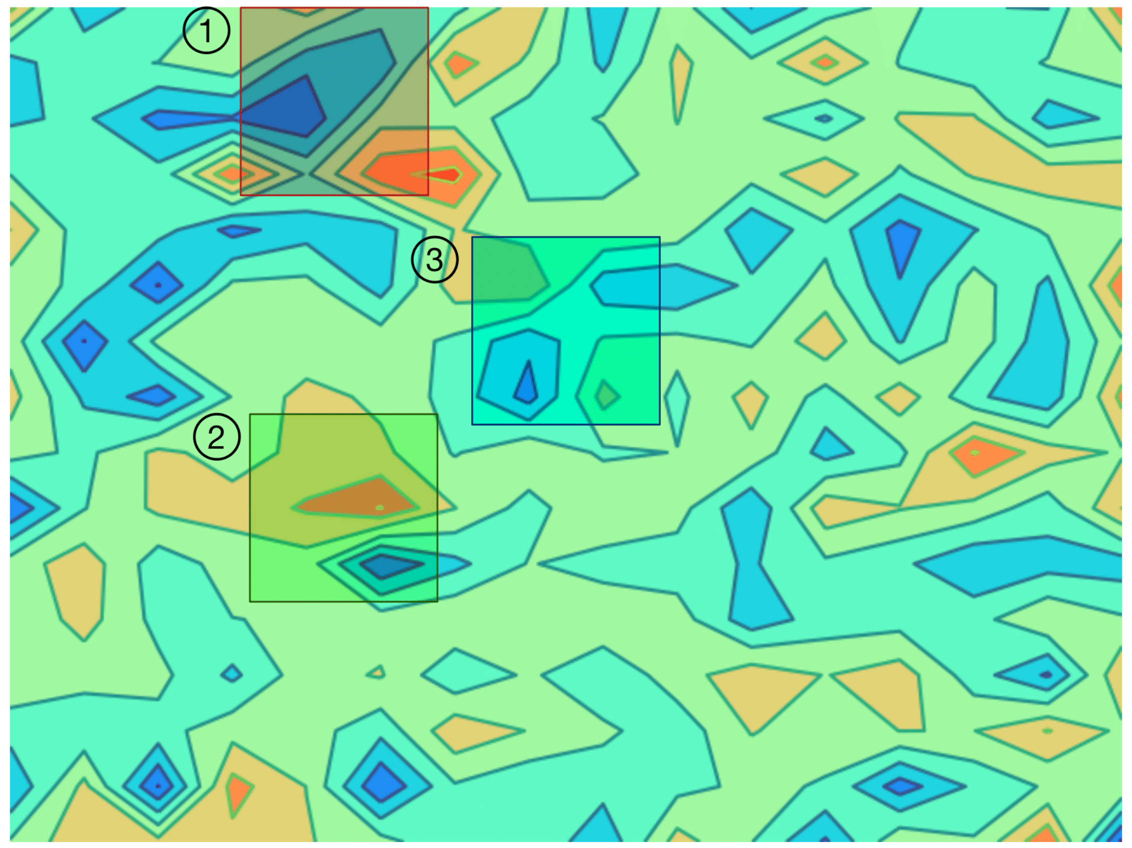
(b) Cross-Entropy Loss on 18-Label RAN

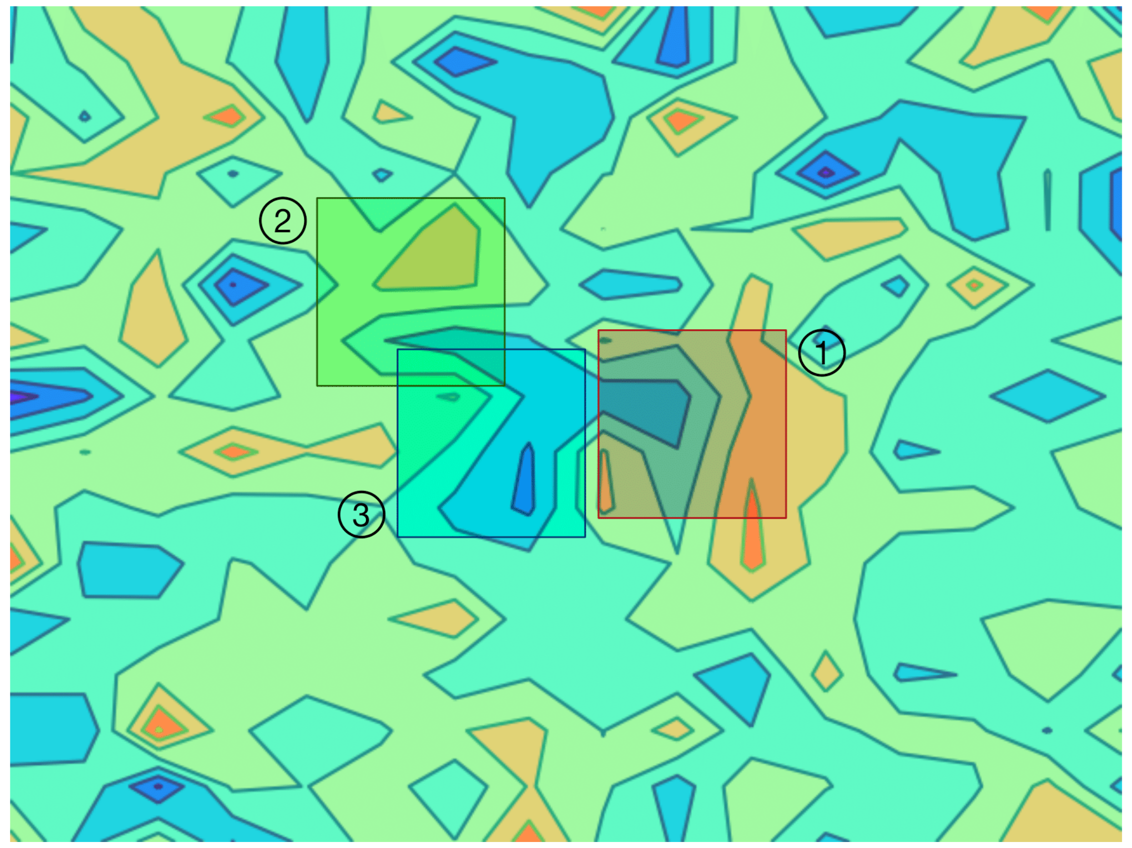
(c) Cross-Entropy Loss on 36-Label RAN

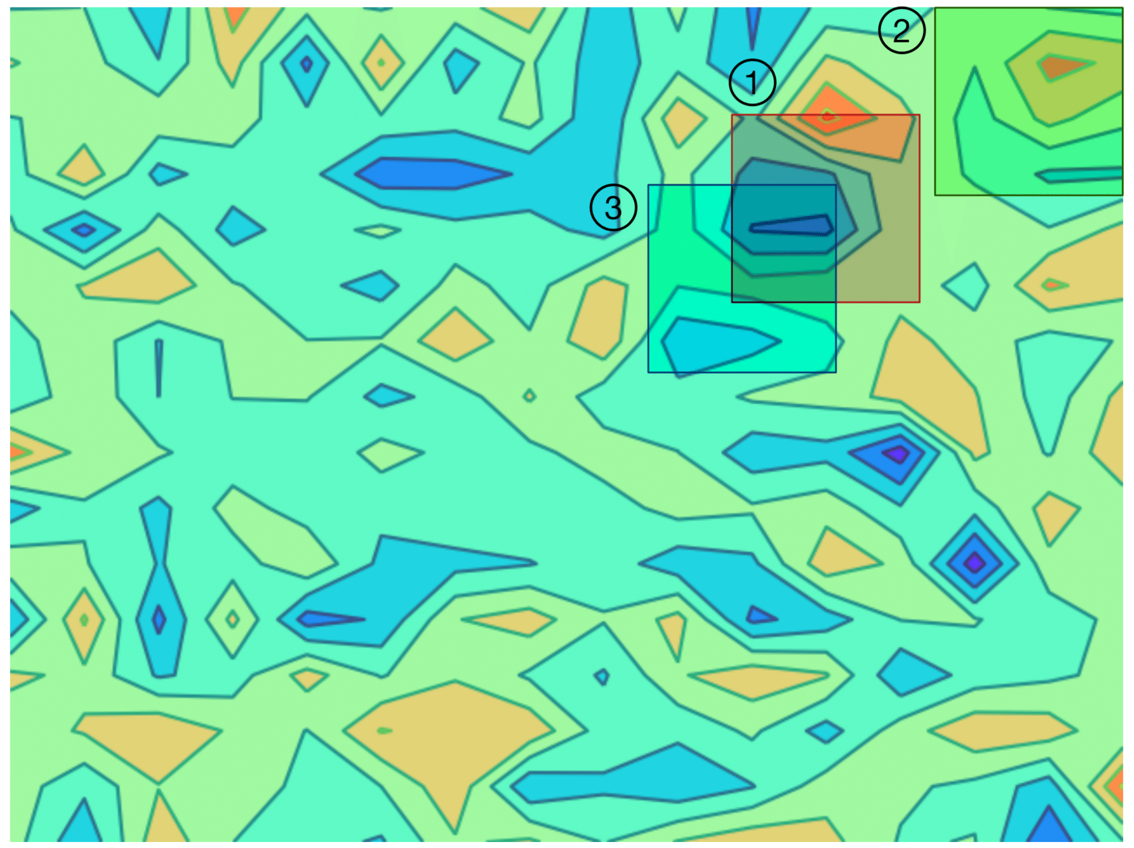
Experimental Results on RAN

Exceptional Performance on Unsupervised Learning



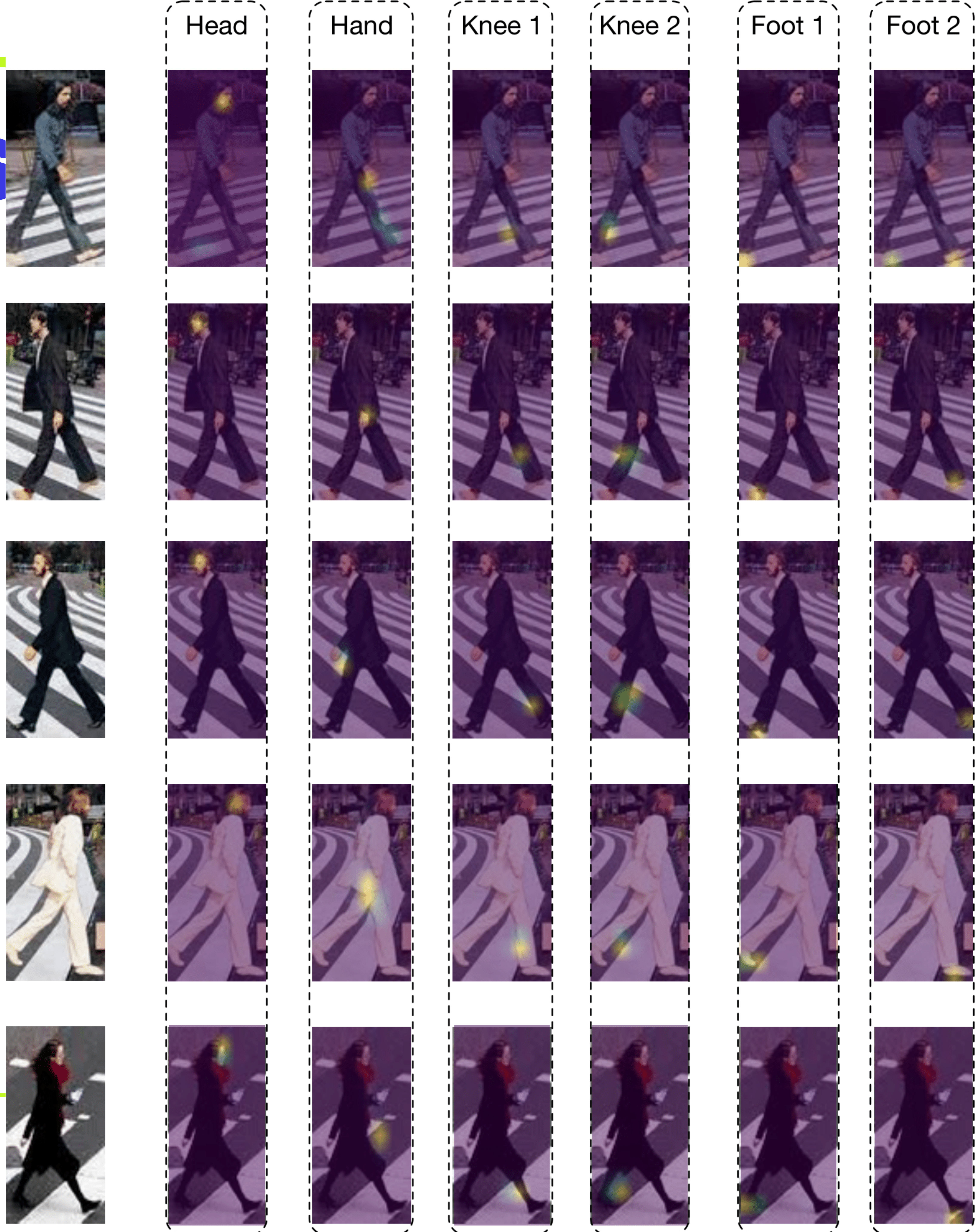




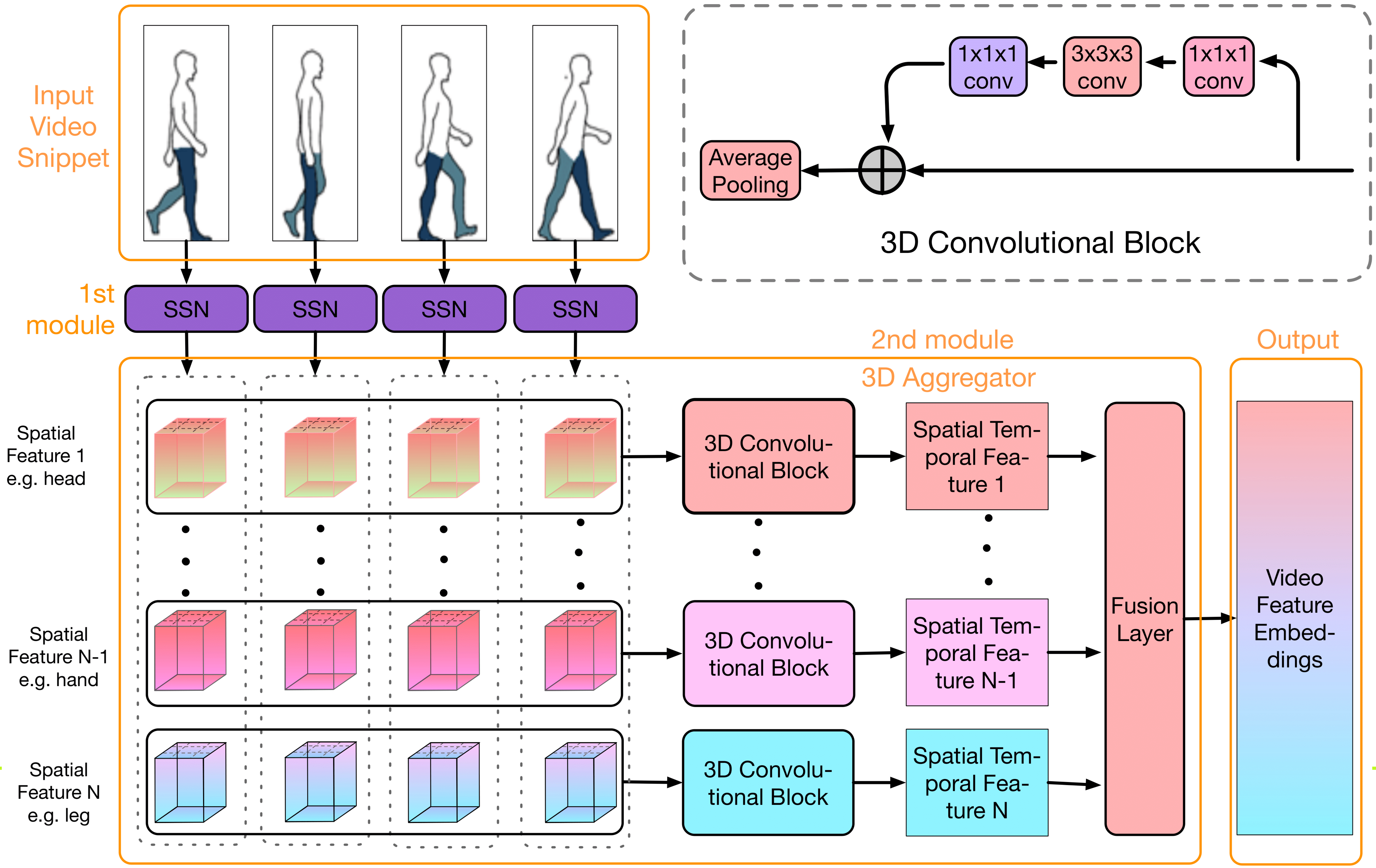


Experimental Results

Impressive Results on Semi-Supervised Learning



3D Convolutional Networks



Objection Function

Triplet Loss + RAN Loss

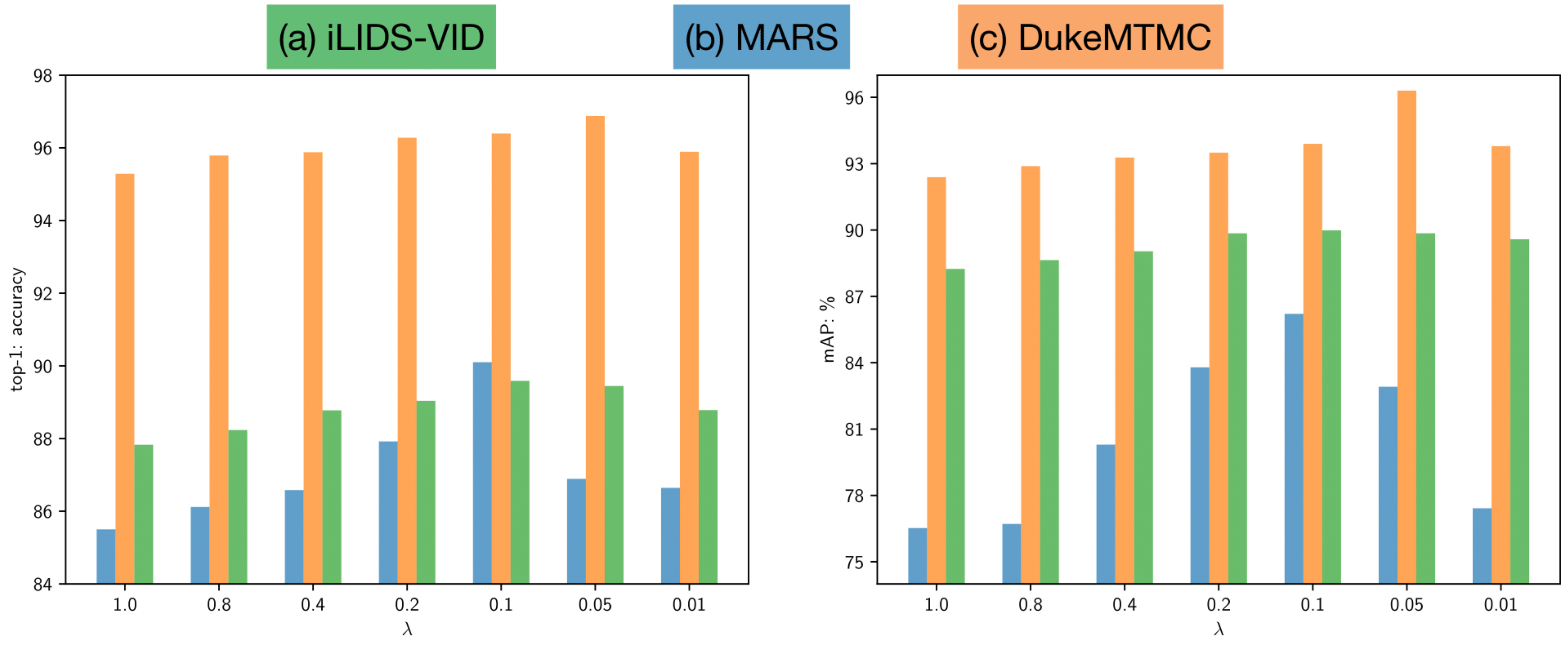
$$\mathcal{L}_{tri} = \sum_{i=1}^{B}[m + \max_{f_p \in S_i^+} \frac{||f_i - f_p||_2}{\sqrt{d}} - \min_{f_n \in S_i^-} \frac{||f_i - f_n||_2}{\sqrt{d}}]_+$$

$$\mathcal{L} = \mathcal{L}_{tri} + \lambda \cdot \mathcal{L}_{RAN}$$

Ablation Study

Learning Strategy

Learning Strategy	iLIDS		MARS		DukeMTMC	
	top-1	mAP	top-1	mAP	top-1	mAP
Supervise	73.4	75.8	69.8	61.1	86.3	79.6
UnSuperv.	83.1	84.2	82.4	67.5	89.9	86.2
Semi-Sup.	**88.9**	**89.2**	**90.1**	**86.2**	**96.8**	**96.3**



With or Without Weight Sharing

Attention Classifiers	iLIDS		MARS		DukeMTMC	
	top-1	mAP	top-1	mAP	top-1	mAP
NonSha.	69.4	73.2	72.3	70.9	84.9	71.2
Sharing	**88.9**	**89.2**	**90.1**	**86.2**	**96.8**	**96.3**

Comparison with State-of-the-arts

Methods	top-1	top-5	top-10
LFDA	32.9	68.5	82.2
KISSME	36.5	67.8	78.8
LADF	39.0	76.8	89.0
STF3D	44.3	71.7	83.7
TDL	56.3	87.6	95.6
MARS	53.0	81.4	-
SeeForest	55.2	86.5	91.0
CNN+RNN	58.0	84.0	91.0
Seq-Decision	60.2	84.7	91.7
ASTPN	62.0	86.0	94.0
QAN	68.0	86.8	95.4
RQEN	77.1	93.2	97.7
STAN	80.2	-	-
Snippet	79.8	91.8	-
Snippet+OF	85.4	96.7	**98.8**
VRSTC	83.4	95.5	97.7
AP3D	86.7	-	-
SSN3D	**88.9**	**97.3**	**98.8**

iLIDS-VID

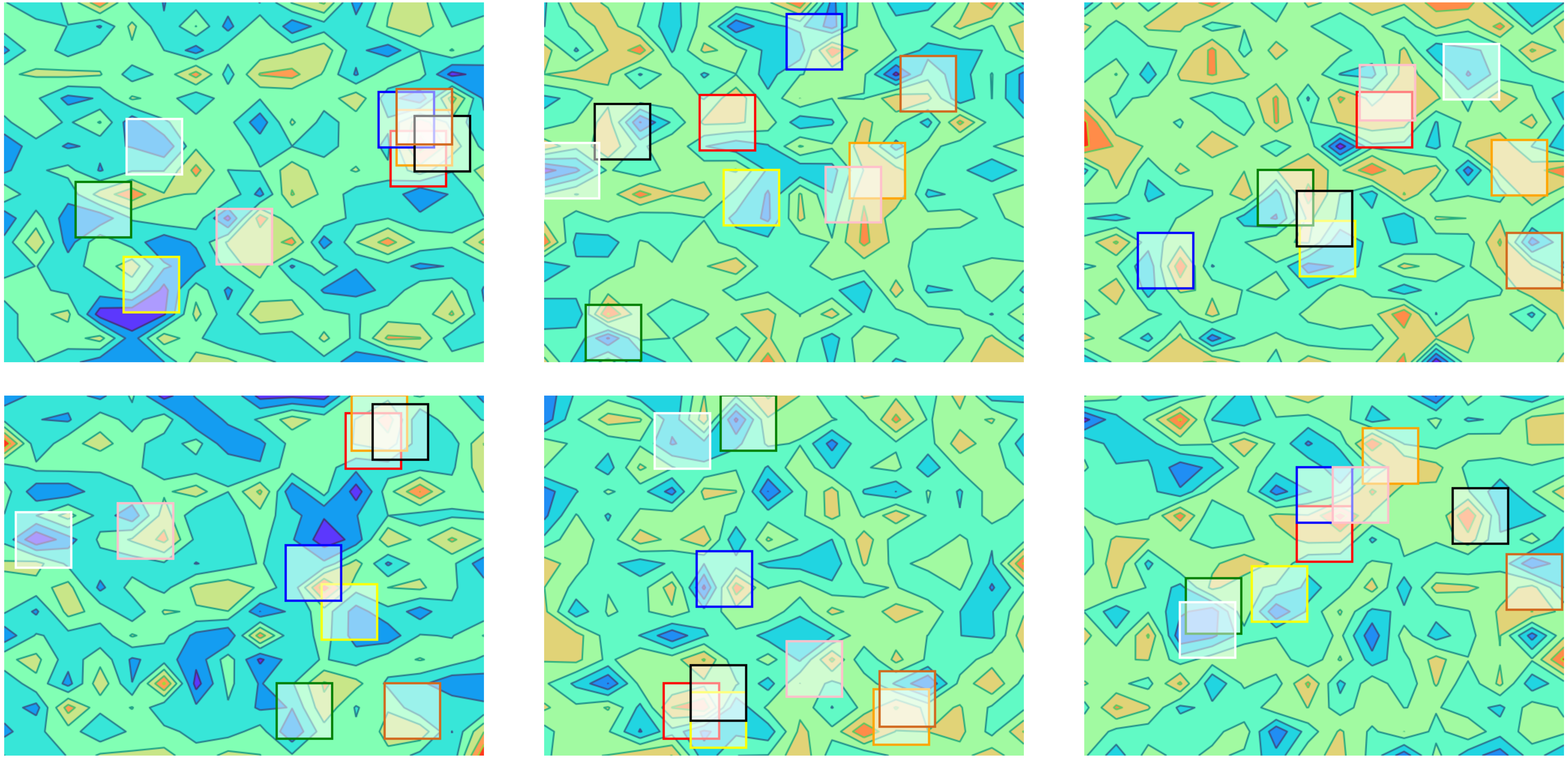
Methods	top-1	top-5	top-10	mAP
Mars	68.3	82.6	89.4	49.3
SeeForest	70.6	90.0	97.6	50.7
Seq-Decision	71.2	85.7	91.8	-
Latent Parts	71.8	86.6	93.0	56.1
QAN	73.7	84.9	91.6	51.7
K-reciprocal	73.9	-	-	68.5
RQEN	77.8	88.8	94.3	71.7
TriNet	79.8	91.3	-	67.7
EUG	80.8	92.1	96.1	67.4
STAN	82.3	-	-	65.8
Snippet	81.2	92.1	-	69.4
Snippet+OF	86.3	94.7	**98.2**	76.1
VRSTC	88.5	96.5	97.4	82.3
AP3D	**90.1**	-	-	85.1
SSN3D	**90.1**	**96.6**	98.0	**86.2**

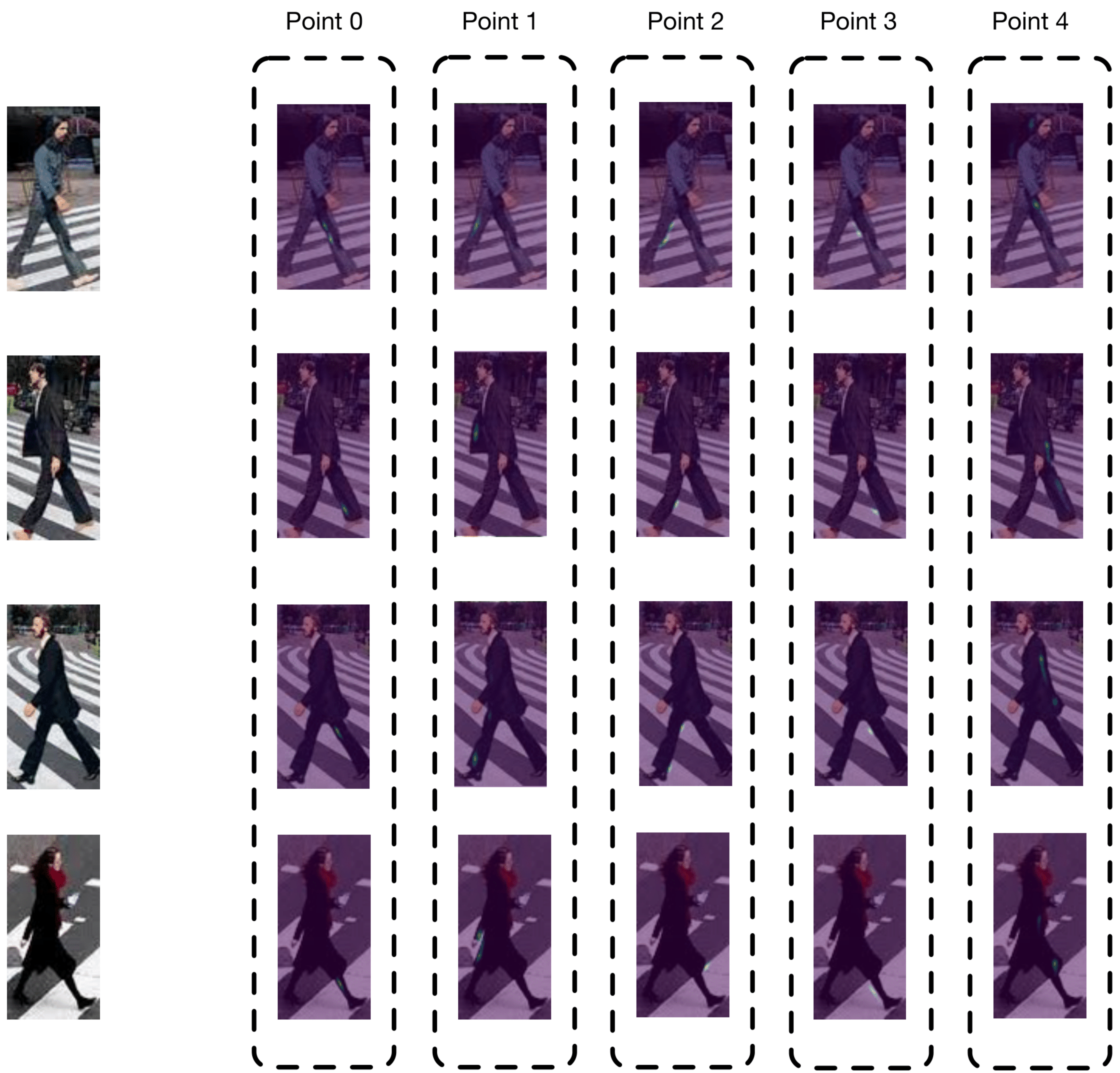
MARS

Methods	top-1	top-5	top-10	mAP
EUG	83.6	94.6	97.6	78.3
VRSTC	95.0	**99.1**	**99.4**	93.5
AP3D	96.3	-	-	95.6
SSN3D	**96.8**	98.6	**99.4**	**96.3**

DukeMRMC

More Results on Amber Abstracts




Point 0
Point 1
Point 2
Point 3
Point 4

Thanks for listening