Seminar 02 - 13 May 2023

OBJECT TRACKING

Computer Vision

Instructors:
Phạm Minh Hoàng
Nguyễn Trọng Việt
Võ Hoài Việt

Team Members: Group03
19127039 – Trần Hoàng Kim
20127578 – Phan Trí Nguyên

Agenda

01. Problem Statement

02. Related Works

03. Methodology

04. Experiments

05. Video Demo

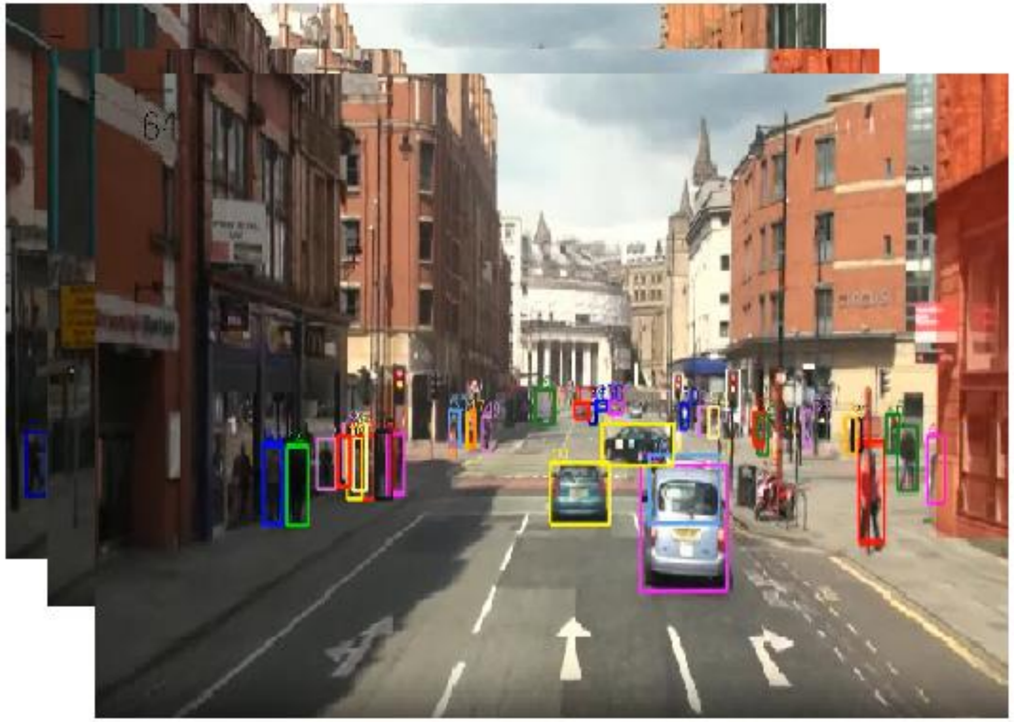
06. Conclusion

01. Problem Statement

01. Problem Statement

Input: sequence of video contain objects (human, vehicle, animal, etc)

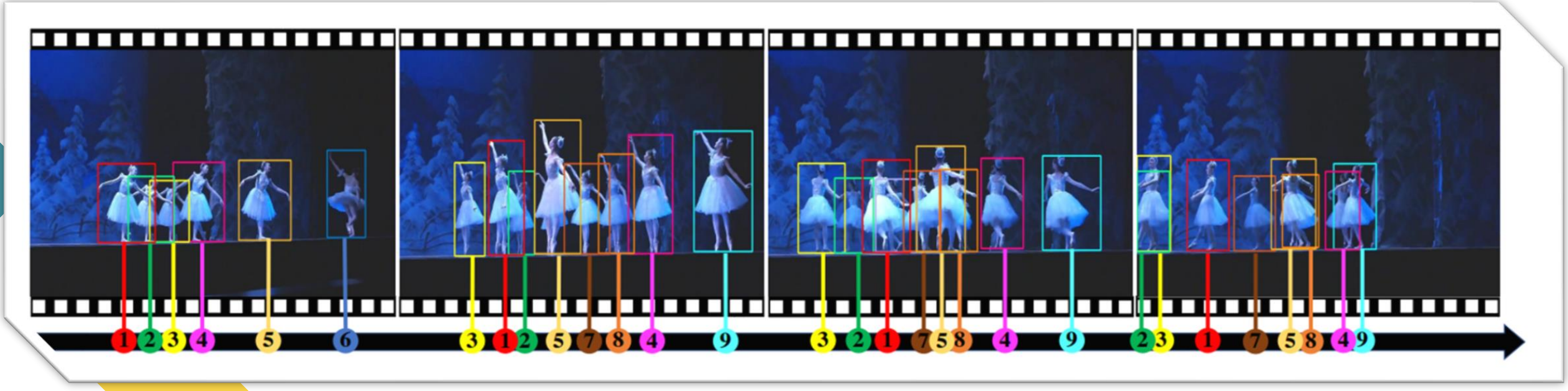
Output: Objects of each frame include bounding boxes and identifications



New Challenge

- Large-scale data for tracking, where objects have similar appearance, diverse motion.

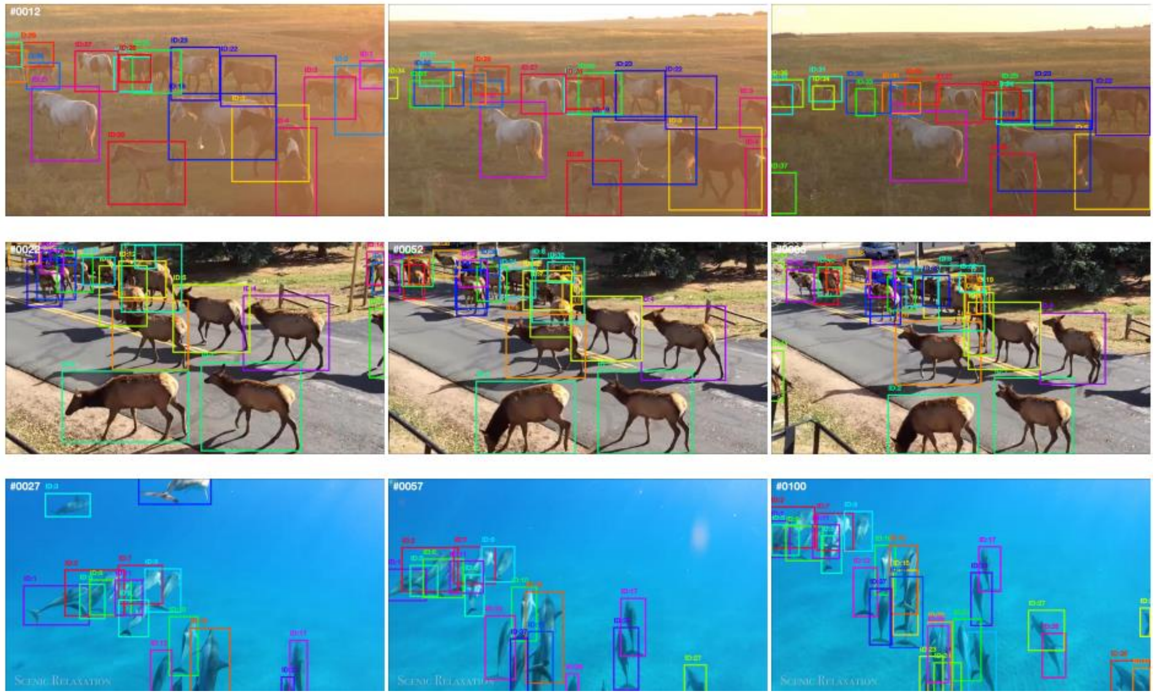
- The numbers below show their identifications which experience frequent relative position switches and occlusions.



Peize Sun, Jinkun Cao(2022) DanceTrack: Multi-Object Tracking in Uniform Appearance and Diverse Motion

New Challenge

- Kind of animals have high similarity appearance which easily causes switching identities



[Libo Zhang, Junyuan Gao, Zhen Xiao, Heng Fan (2022) AnimalTrack: A Benchmark for Multi-Animal Tracking in the Wild]

02.
Related Works

02. Related Works

Two-stage method		One-stage method	
SORT: Simple Online Realtime Object Tracking	**DeepSORT**: SORT with a Deep Association Metric	**TrackFormer**: Multi-Object Tracking with Transformers	**QDTrack**: Quasi-Dense Similarity Learning for Multiple Object Tracking

Two-Stage (Tracking-by-detection)

One characteristic of the class of Tracking-by-detection algorithms is to separate object detection as a separate problem and attempt to optimize the results in this task.

The next step is to find a way to link the bounding boxes obtained in each frame and assign an ID to each object.
Therefore, we have a processing pipeline for each new frame as follows:

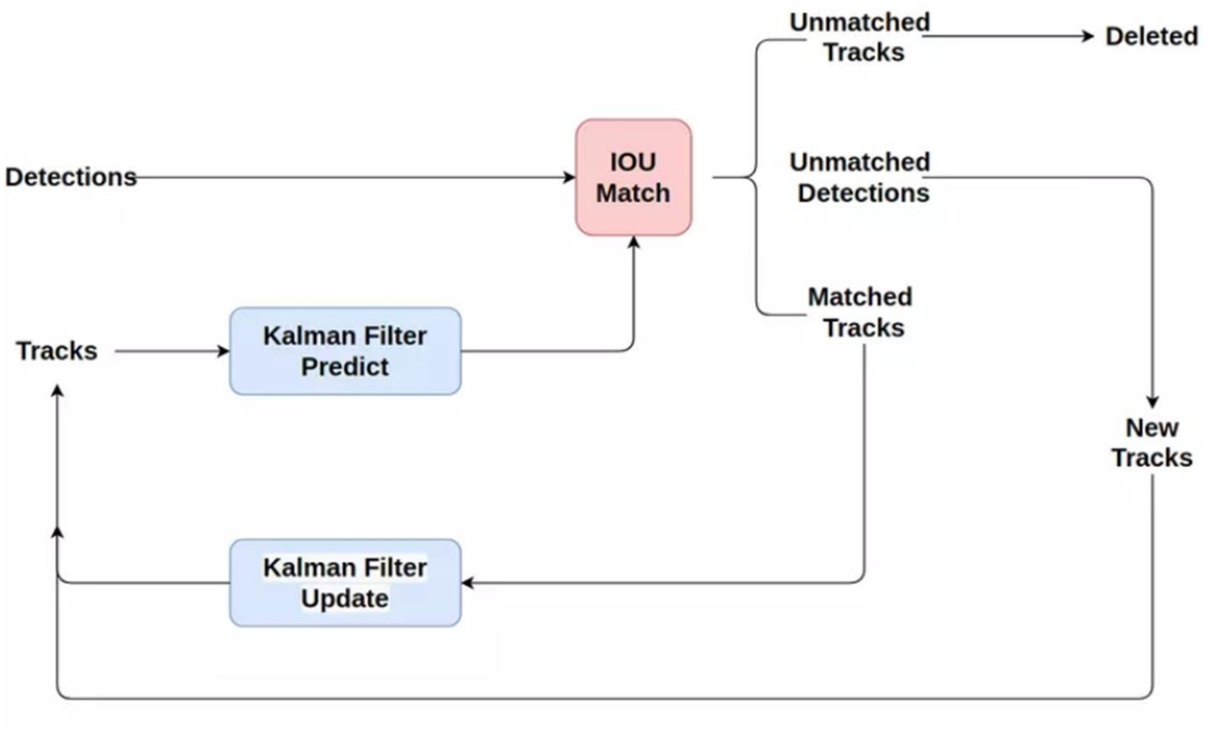
Method 1

- SORT - Simple Online Realtime Object Tracking

Method 2

- DeepSORT - SORT with a Deep Association Metric

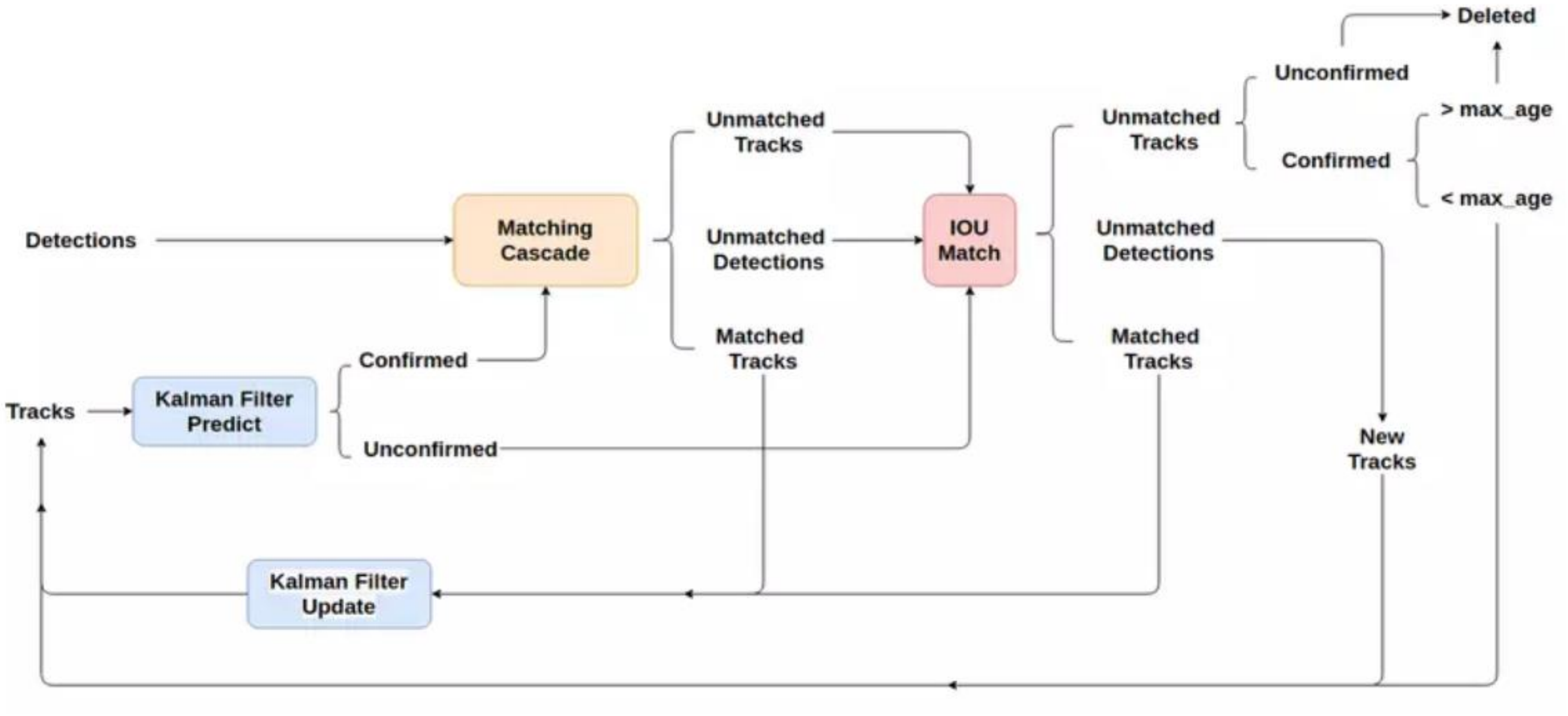
SORT - Simple Online Realtime Object Tracking



[Alex Bewley, Zongyuan Ge (2016): Simple Online and Realtime Tracking]

- SORT uses the Kalman filter to predict the position of tracks in the next frame based on the past frames.
- And then matching detections (from object detection) and tracks by Hungarian matching algorithm (matching IoU).

DeepSORT - SORT with a Deep Association Metric



[Nicolai Wojke, Alex Bewley (2017) Simple Online and Realtime Tracking with a Deep Association Metric]

DeepSort is the improvement of SORT by additional matching appearances.

Weakness of two-stage methods

- Completely depend on object detection.
- Slow speed is a problem in processing real-time tracking.

=> We utilize **one-stage method** in this seminar to improve this weaknesses.

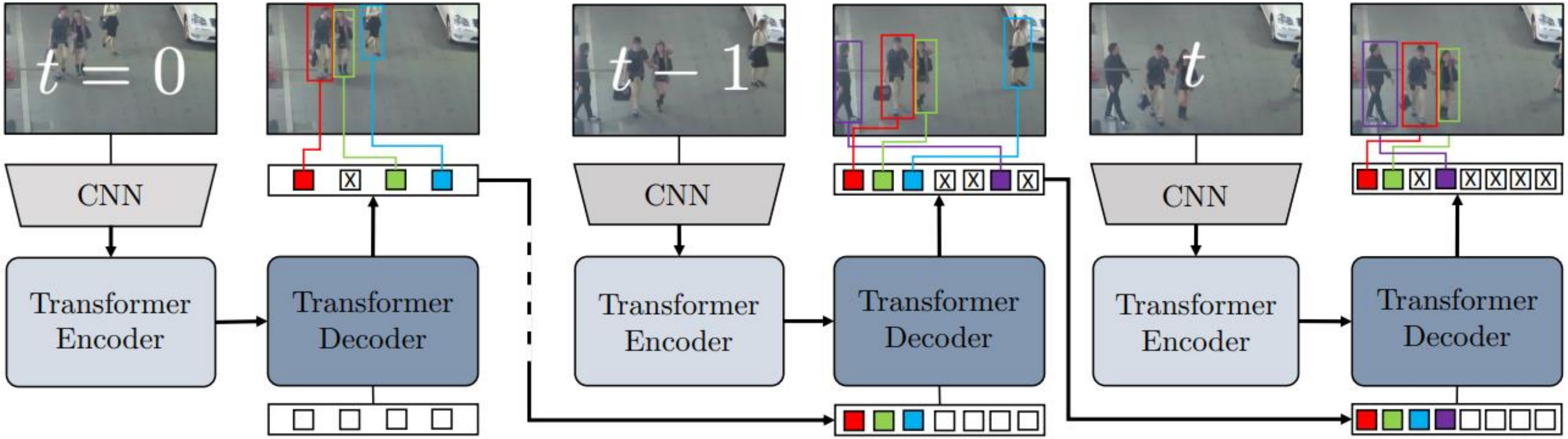
Although one-stage method has a significant impact on practical application through its real-time capability, this method is complicated.

One-Stage (End-to-end)

Joint detection and tracking methods are classified as one-stage MOT, in which the detection and tracking steps are simultaneously produced in a **single network.**

In this category, object detection can be modeled within a single network with **re-ID feature extraction** or **motion features**.

TrackFormer: Multi-Object Tracking with Transformers

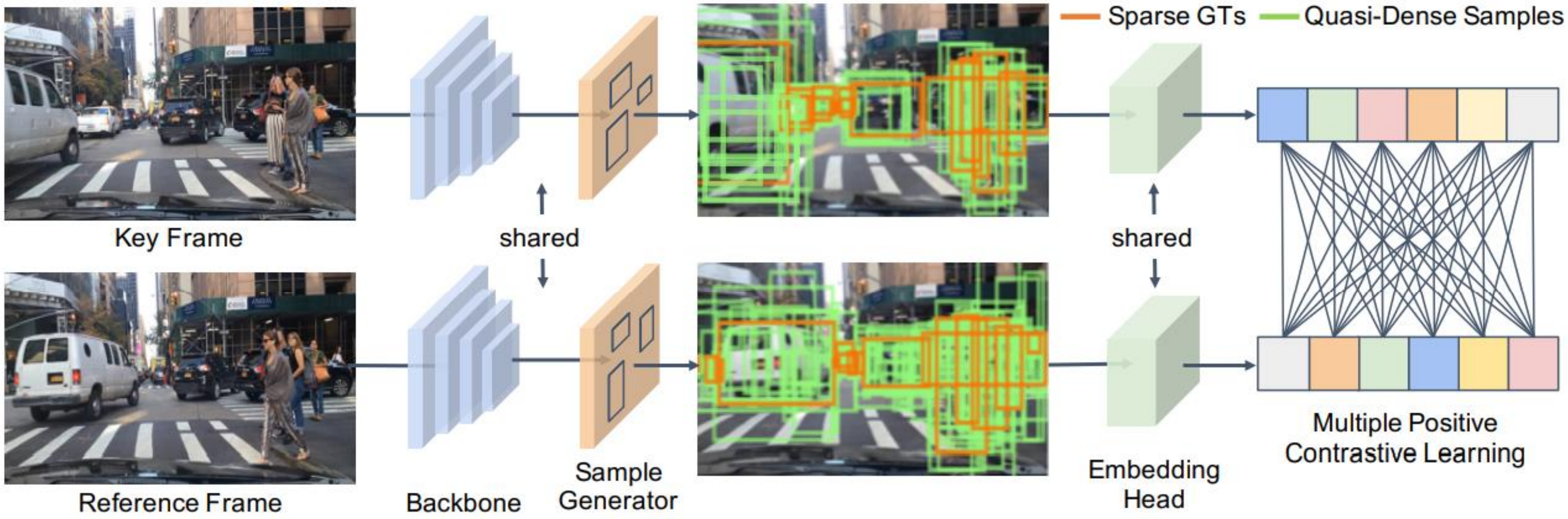


Tim Meinhardt, Alexander Kirillov(2022) TrackFormer: Multi-Object Tracking with Transformers

TrackFormer utilizes Transformer with two main taskes:
- Transform object queries (white squares) to new tracks queries or background (like object detection task)
- Transform track queries (color squares) from previous frames to tracks in current frame (like matching task)

Quasi-Dense Similarity Learning for Multiple Object Tracking

Training pipeline

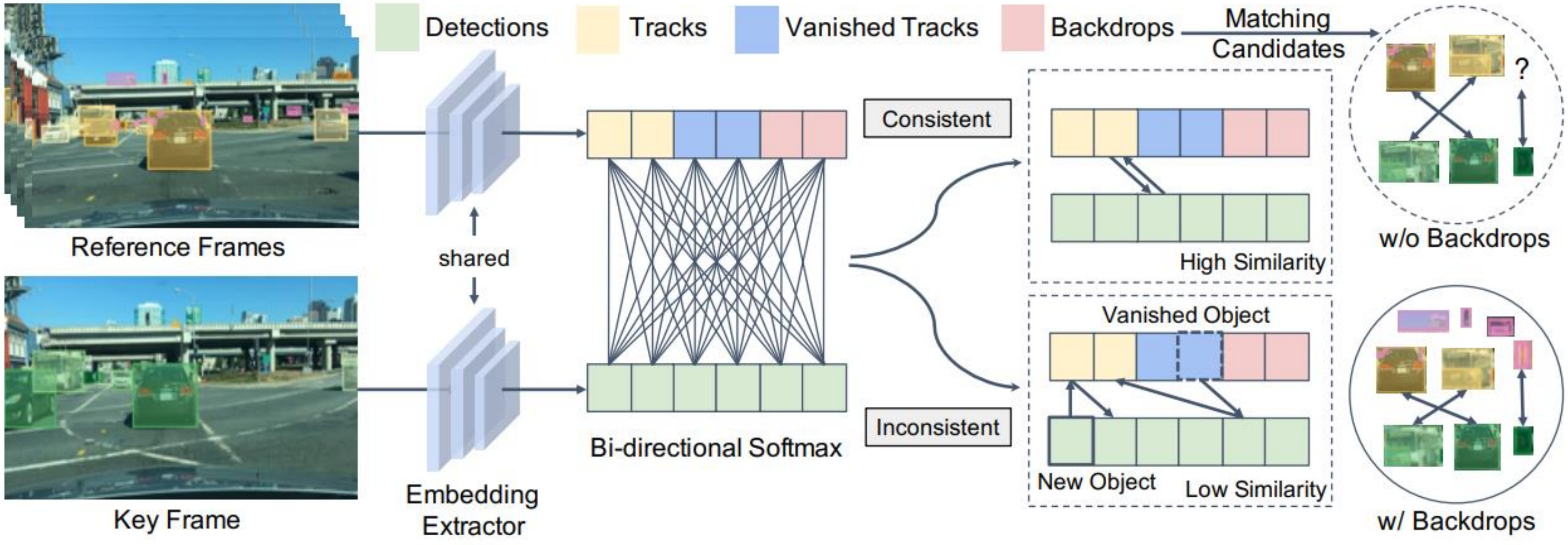


Jiangmiao Pang, Linlu Qiu, Xia Li (2021) Quasi-Dense Similarity Learning for Multiple Object Tracking

The training process creates a powerful re-ID network by **quasi-dense similarity learning** which learn the feature embedding space that can **associate identical objects and distinguish different objects for online multiple object tracking**

Quasi-Dense Similarity Learning for Multiple Object Tracking

Testing pipeline



Jiangmiao Pang, Linlu Qiu, Xia Li (2021) Quasi-Dense Similarity Learning for Multiple Object Tracking

The testing process utilizes re-ID trained network to match proposal objects in the current/key frame and past/reference frames.

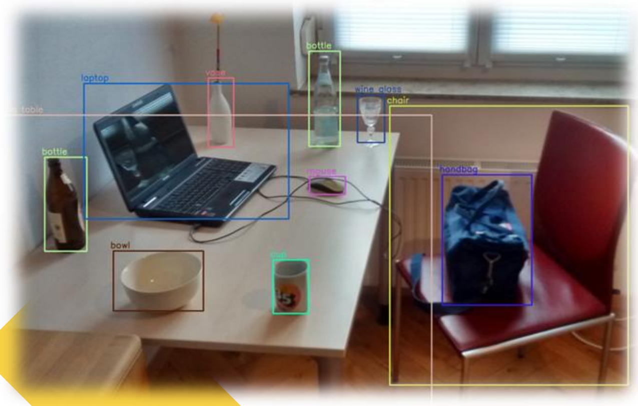
03.
Methodology

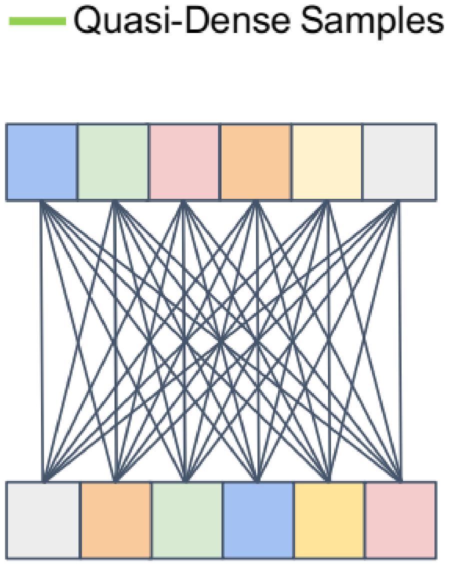
Quasi-Dense Similarity Learning for Multiple Object Tracking

Object detection

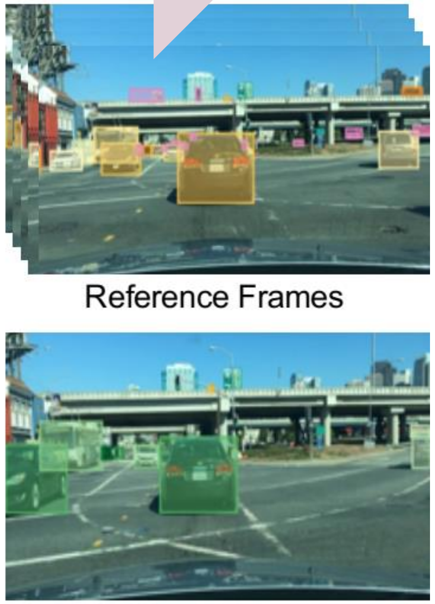
Quasi-dense similarity learning

Object association





Multiple Positive Contrastive Learning



Reference Frames

Key Frame

03. Methodology

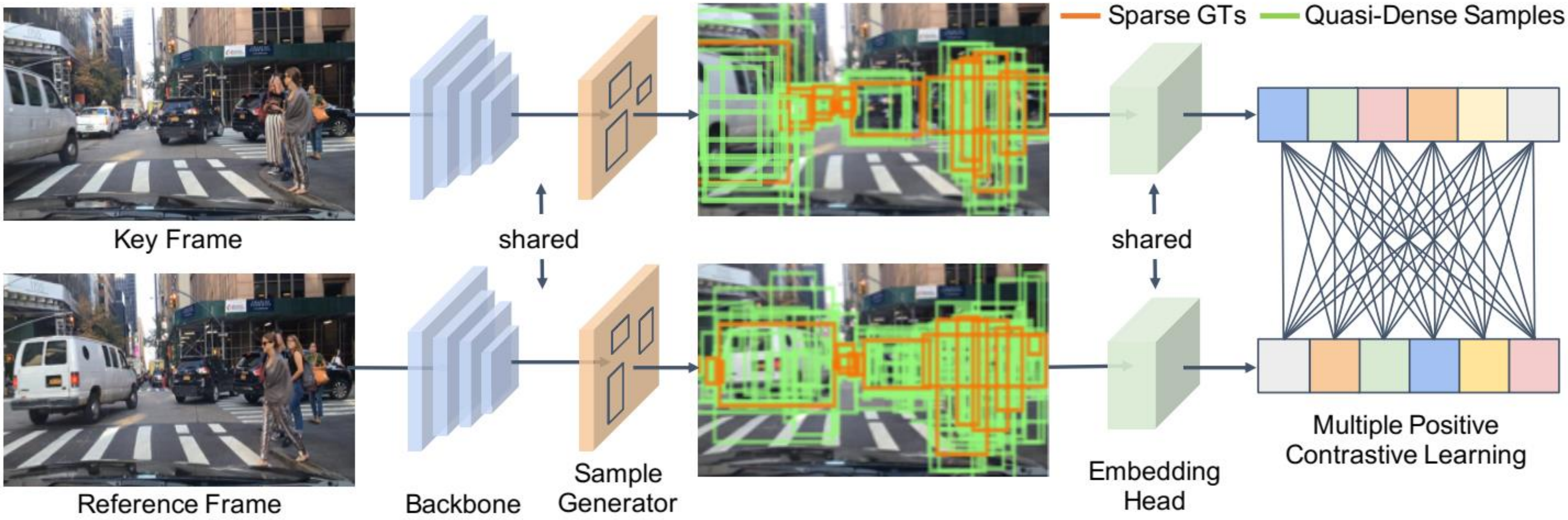
OBJECT DETECTION

- Faster R-CNN
- Feature Pyramid Network

A multi-task loss function:

$$\mathcal{L}_{det} = \underbrace{\mathcal{L}_{rpn}}_{\text{RPN loss}} + \underbrace{\lambda_1 \mathcal{L}_{cls}}_{\text{Classification Loss}} + \underbrace{\lambda_2 \mathcal{L}_{reg}}_{\text{Regression Loss}} \quad (1)$$

03. Methodology
Quasi-dense similarity learning



[Jiangmiao Pang, Linlu Qiu, Xia Li, Haofeng Chen, Qi Li, Trevor Darrell, Fisher Yu (2021) Quasi-Dense Similarity Learning for Multiple Object Tracking]

“

$$\mathcal{L}_{embed} = \log[1 + \sum_{k^+} \sum_{k^-} \exp(v.k^- - v.k^+)] \quad (2)$$

An auxiliary loss:

$$\mathcal{L}_{aux} = \left(\frac{v.k}{||v||.||k||} - c\right)^2 \quad (3)$$

- ✓ Aim: constrain the logit magnitude and cosine similarity

Quasi-dense similarity learning

A multi-task loss function:

$$\mathcal{L}_{det} = \mathcal{L}_{rpn} + \lambda_1 \mathcal{L}_{cls} + \lambda_2 \mathcal{L}_{reg}$$

An auxiliary loss:

$$\mathcal{L}_{aux} = \left(\frac{v.k}{||v||.||k||} - c\right)^2$$

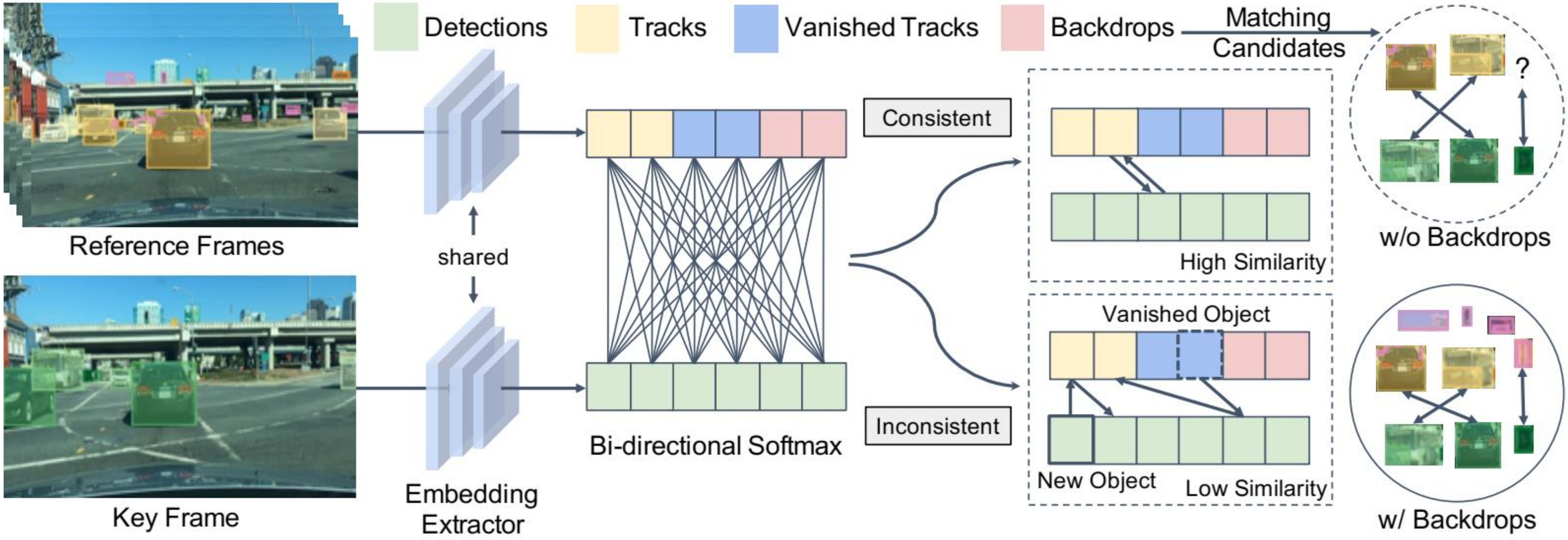
Dense matching:

$$\mathcal{L}_{embed} = \log[1 + \sum_{k^+} \sum_{k^-} \exp(v.k^- - v.k^+)]$$

The entire network:

$$\mathcal{L} = \mathcal{L}_{det} + \gamma_1 \mathcal{L}_{embed} + \gamma_2 \mathcal{L}_{aux}$$

03. Methodology
Object association



Jiangmiao Pang, Linlu Qiu, Xia Li, Haofeng Chen, Qi Li, Trevor Darrell, Fisher Yu (2021) Quasi-Dense Similarity Learning for Multiple Object Tracking

Bi-directional softmax

$$f(i,j) = \left[\frac{\exp(n_i . m_j)}{\sum_{k=0}^{M-1} \exp(n_i . m_k)} + \frac{\exp(n_i . m_j)}{\sum_{k=0}^{N-1} \exp(n_k . m_j)}\right] / 2$$

04. Experiments

04. Experiments Datasets

MOT16 and MOT17

Training Set: 7 videos - 5,316 images

Testing Set: 7 videos - 5,919 images

Video frame rate: 14 - 30 FPS

MOT16 MOT17

Anton Milan, Laura Leal-Taixé, Ian Reid, Stefan Roth, and Konrad Schindler, MOT16: A Benchmark for Multi-Object Tracking, arXiv:1603.00831v2 [cs.CV] 3 May 2016

04. Experiments Datasets

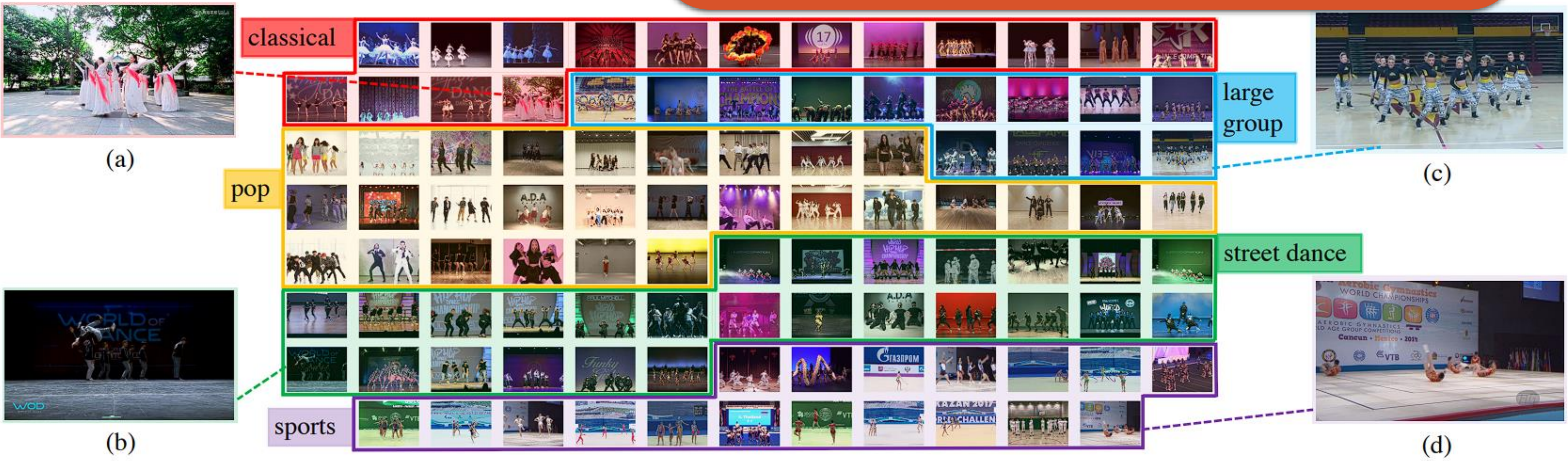
DanceTrack

Total: 100 videos – 105,855 images

Training Set: 40 videos

Video frame rate: 20 FPS.

Testing Set: 35 videos



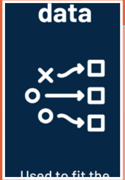
Peize Sun, Jinkun Cao, Yi Jiang, Zehuan Yuan, Song Bai, Kris Kitani, Ping Luo (2022) DanceTrack: Multi-Object Tracking in Uniform Appearance and Diverse Motion

04. Experiments
Datasets

AnimalTrack

Total: 58 video

Video frame rate:
30 FPS.



Training Set:
32 videos

Testing Set:
26 videos

Libo Zhang, Junyuan Gao, Zhen Xiao, Heng Fan (2022) AnimalTrack: A Benchmark for Multi-Animal Tracking in the Wild

04. Experiments
Metrics for evaluation

MOTA

MOTA [17, 18]

It measures the overall accuracy of both the tracker and detection.

It deals with both tracker output and detection output.

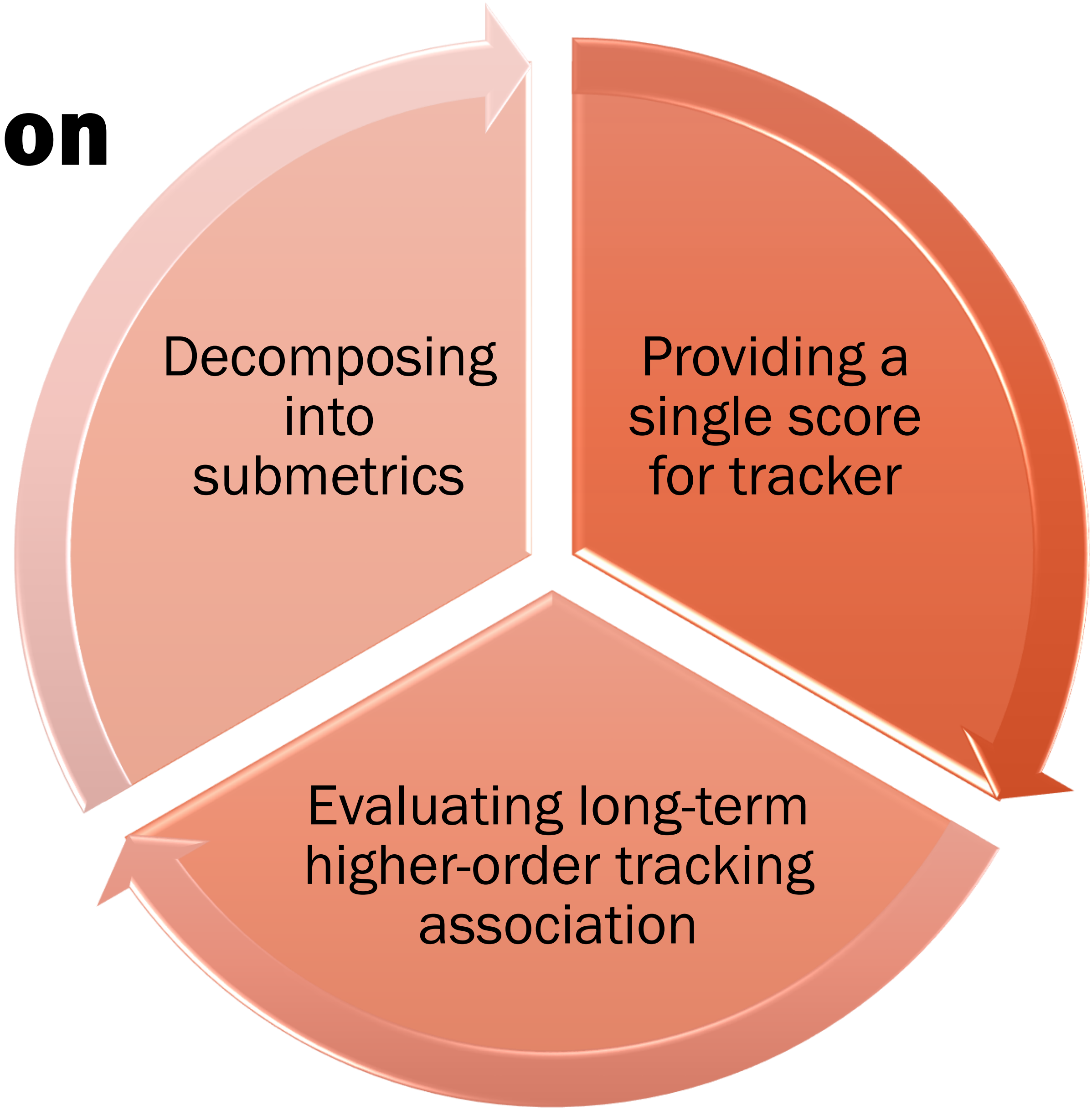
MOTA Calculation

$$MOTA = 1 - \frac{\sum_t FN_t + FP_t + IDS_t}{\sum_t GT_t}$$

04. Experiments
Metrics for evaluation

HOTA

(Higher Order Tracking Accuracy)



Measuring Association

$$\mathrm{TPA}(c) = \{k\},$$
$$k \in \{\mathrm{TP} \mid \mathrm{prID}(k) = \mathrm{prID}(c) \land \mathrm{gtID}(k) = \mathrm{gtID}(c)\} \quad \mathbf{(1)}$$

$$\mathrm{FNA}(c) = \{k\},$$
$$k \in \{\mathrm{TP} \mid \mathrm{prID}(k) \neq \mathrm{prID}(c) \land \mathrm{gtID}(k) = \mathrm{gtID}(c)\} \quad \mathbf{(2)}$$
$$\cup \ \{\mathrm{FN} \mid \mathrm{gtID}(k) = \mathrm{gtID}(c)\}$$

$$\mathrm{FPA}(c) = \{k\},$$
$$k \in \{\mathrm{TP} \mid \mathrm{prID}(k) = \mathrm{prID}(c) \land \mathrm{gtID}(k) \neq \mathrm{gtID}(c) \quad \mathbf{(3)}$$
$$\cup \ \{\mathrm{FP} \mid \mathrm{prID}(k) = \mathrm{prID}(c)\}$$

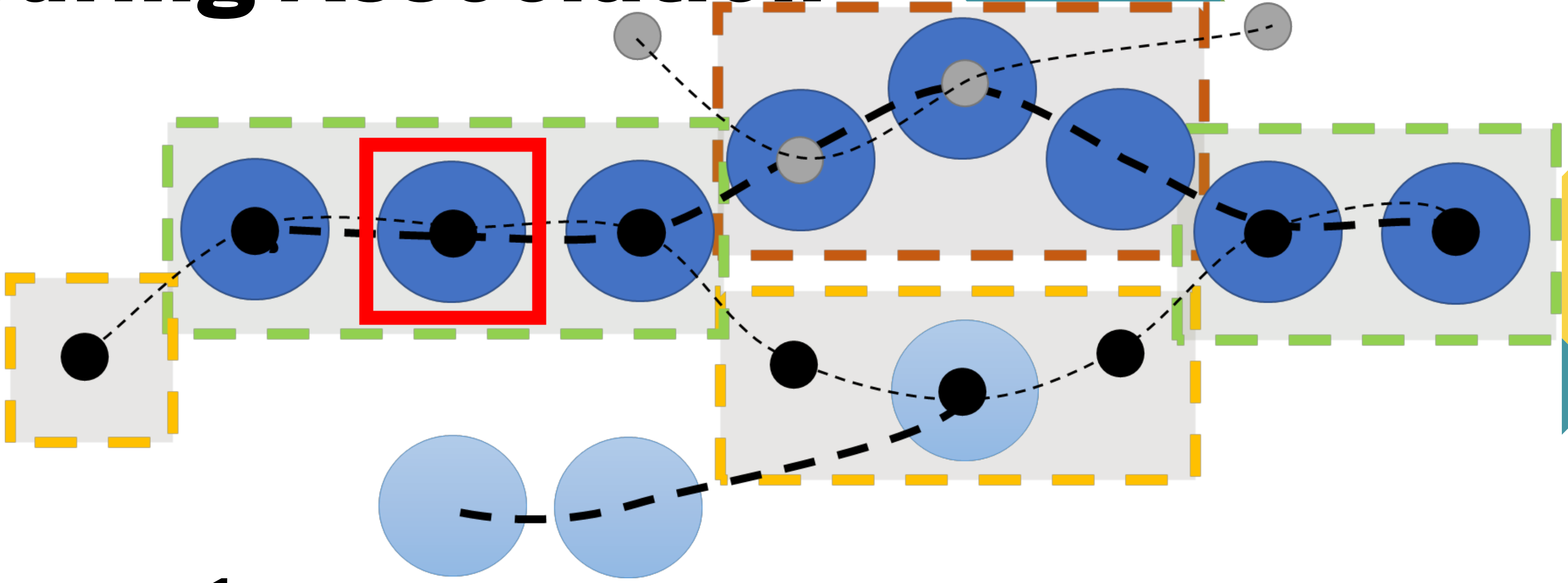
Measuring Association

A particular localisation threshold α for a scoring function from **(4)**, **(5)** and **(6)** equations:

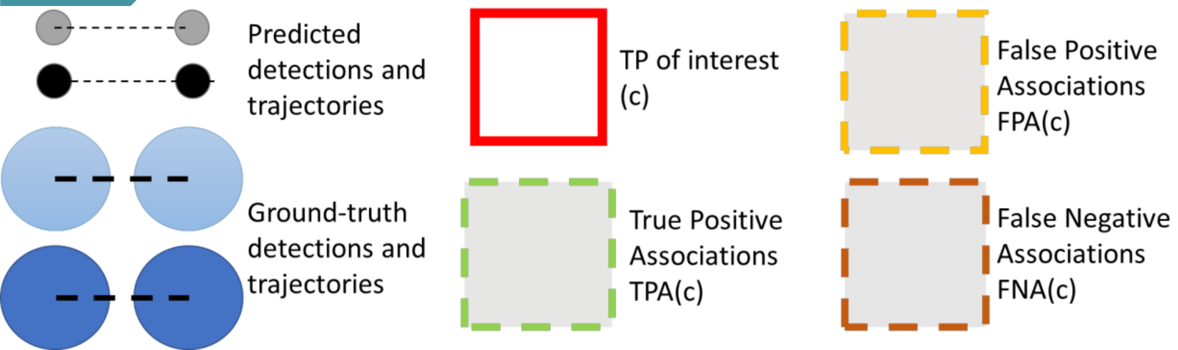
$$\mathcal{A}(c) = \frac{|\mathrm{TPA}(c)|}{|\mathrm{TPA}(c)| + |\mathrm{FNA}(c)| + |\mathrm{FPA}(c)|}$$

$$HOTA_{\alpha} = \sqrt{\frac{\sum_{c \in \{TP\}} \mathcal{A}(c)}{|TP| + |FN| + |FP|}}$$

Measuring Association



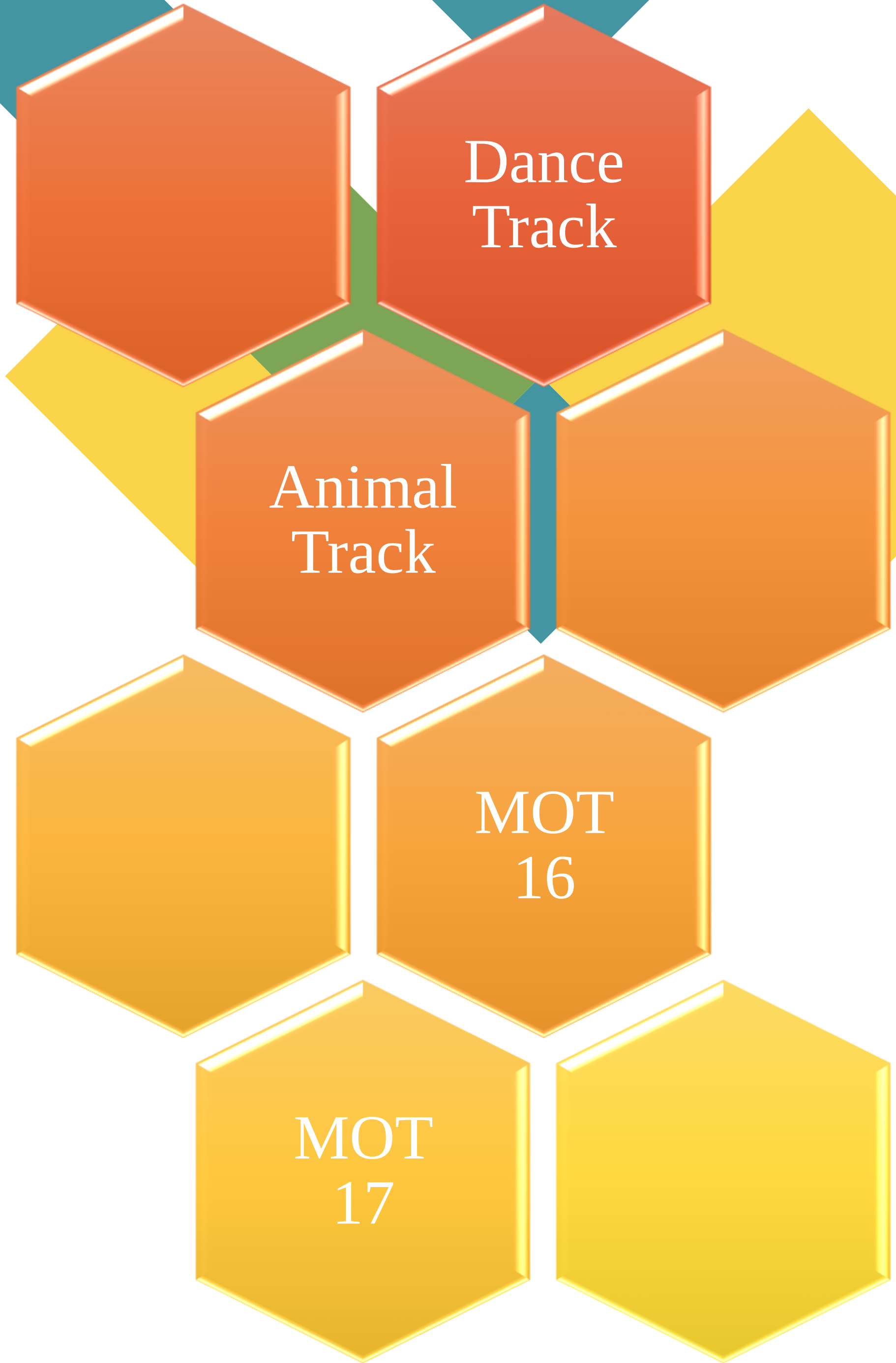
$$HOTA = \int_{0}^{1} HOTA_{\alpha} \approx \frac{1}{19} \sum_{\alpha \in \{0.05, 0.1, \ldots, 0.9, 0.95\}} HOTA_{\alpha}$$



04. Experiments Results

“

Let’s make some comparisons !



04. Experiments - Results

Methods	DanceTrack (Proposed Dataset)				
	HOTA	DetA	AssA	MOTA	IDF1
CenterTrack[1] [42]	41.8	78.1	22.6	86.8	35.7
FairMOT[1] [41]	39.7	66.7	23.8	82.2	40.8
QDTrack[3] [25]	54.2	80.1	36.8	87.7	50.4
TransTrack[1] [29]	45.5	75.9	27.5	88.4	45.2
TraDes[1] [35]	43.3	74.5	25.4	86.2	41.2
MOTR[2] [39]	54.2	73.5	**40.2**	79.7	51.5
GTR[2] [44]	48.0	72.5	31.9	84.7	50.3
ByteTrack[1] [40]	47.7	71.0	32.1	89.6	53.9
OC-SORT[2] [5]	**55.1**	**80.3**	38.3	**92.0**	**54.6**

DanceTrack: Multi-Object Tracking in Uniform Appearance and Diverse Motion

04. Experiments - Results

Tracker	HOTA	MOTA	IDF1	IDP	IDR	MT	PT	ML↓	FP↓	FN↓	IDs↓	FM↓
SORT [6]	42.8%	55.6%	49.2%	58.5%	42.4%	333	470	301	19,099	86,257	2,530	3,730
IOUTrack [7]	41.6%	55.7%	45.7%	51.9%	40.7%	**388**	454	**262**	25,206	**77,847**	4,639	5,259
DeepSORT [54]	32.8%	41.4%	35.2%	49.7%	27.2%	213	452	439	14,131	124,747	3,503	4,527
JDE [52]	26.8%	27.3%	31.0%	51.0%	22.0%	106	414	584	17,887	155,623	3,187	5,031
FairMOT [62]	30.6%	29.0%	38.8%	62.8%	28.0%	143	462	499	17,653	152,624	2,335	5,447
CenterTrack [63]	9.9%	1.6%	7.0%	8.9%	5.8%	265	423	416	32,050	117,614	89,655	7,583
CTracker [41]	13.8%	14.0%	14.7%	35.2%	9.3%	20	313	771	**13,092**	192,660	3,437	8,019
Tracktor++ [3]	44.2%	**55.2%**	51.0%	58.5%	45.1%	364	472	**268**	25,477	**81,538**	**1,976**	4,149
ByteTrack [61]	40.1%	38.5%	51.2%	**64.9%**	42.3%	310	465	329	31,591	116,587	**1,309**	**3,513**
QDTrack [39]	**47.0%**	**55.7%**	**56.3%**	**65.6%**	**49.3%**	**367**	420	317	22,696	83,057	1,970	5,656
TADAM [23]	32.5%	36.5%	37.2%	44.4%	32.0%	258	**495**	351	41,728	110,048	2,538	4,469
OMC [29]	43.0%	53.4%	50.3%	61.8%	42.4%	324	478	302	**15,910**	92,570	4,938	7,162
Trackformer [37]	31.0%	20.4%	36.5%	40.9%	32.8%	230	**491**	383	70,404	118,724	4,355	**3,725**
TransTrack [48]	**45.4%**	48.3%	**53.4%**	63.4%	**46.1%**	327	416	361	28,553	95,212	1,978	6,459

AnimalTrack: A Benchmark for Multi-Animal Tracking in the Wild

Jiangmiao Pang, (2021) Quasi-Dense Similarity Learning for Multiple Object Tracking

Dataset	Method	MOTA ↑	IDF1 ↑	MOTP ↑	MT ↑	ML ↓	FP ↓	FN ↓	IDs ↓
MOT16	TAP [57]	64.8	**73.5**	78.7	292	164	12980	50635	**571**
	CNNMTT [29]	65.2	62.2	78.4	246	162	6578	55896	946
	POI* [54]	66.1	65.1	**79.5**	258	158	**5061**	55914	3093
	TubeTK_POI* [35]	66.9	62.2	78.5	296	**122**	11544	47502	1236
	CTrackerV1 [37]	67.6	57.2	78.4	250	175	8934	48305	1897
	Ours	**69.8**	67.1	79.0	**316**	150	9861	**44050**	1097
MOT17	Tracktor++v2 [2]	56.3	55.1	78.8	498	831	**8866**	235449	1987
	Lif_T* [20]	60.5	65.6	78.3	637	791	14966	206619	**1189**
	TubeTK* [35]	63.0	58.6	78.3	735	**468**	27060	177483	4137
	CTrackerV1 [37]	66.6	57.4	78.2	759	570	22284	160491	5529
	CenterTrack* [56]	67.8	64.7	78.4	816	579	18498	160332	3039
	Ours	**68.7**	**66.3**	**79.0**	**957**	516	26589	**146643**	3378

QDTrack's Results on MOT16 and MOT17 test set with private detectors

↑ means higher is better

↓ means lower is better

∗ means external data besides COCO and ImageNet is used

05. Video Demo

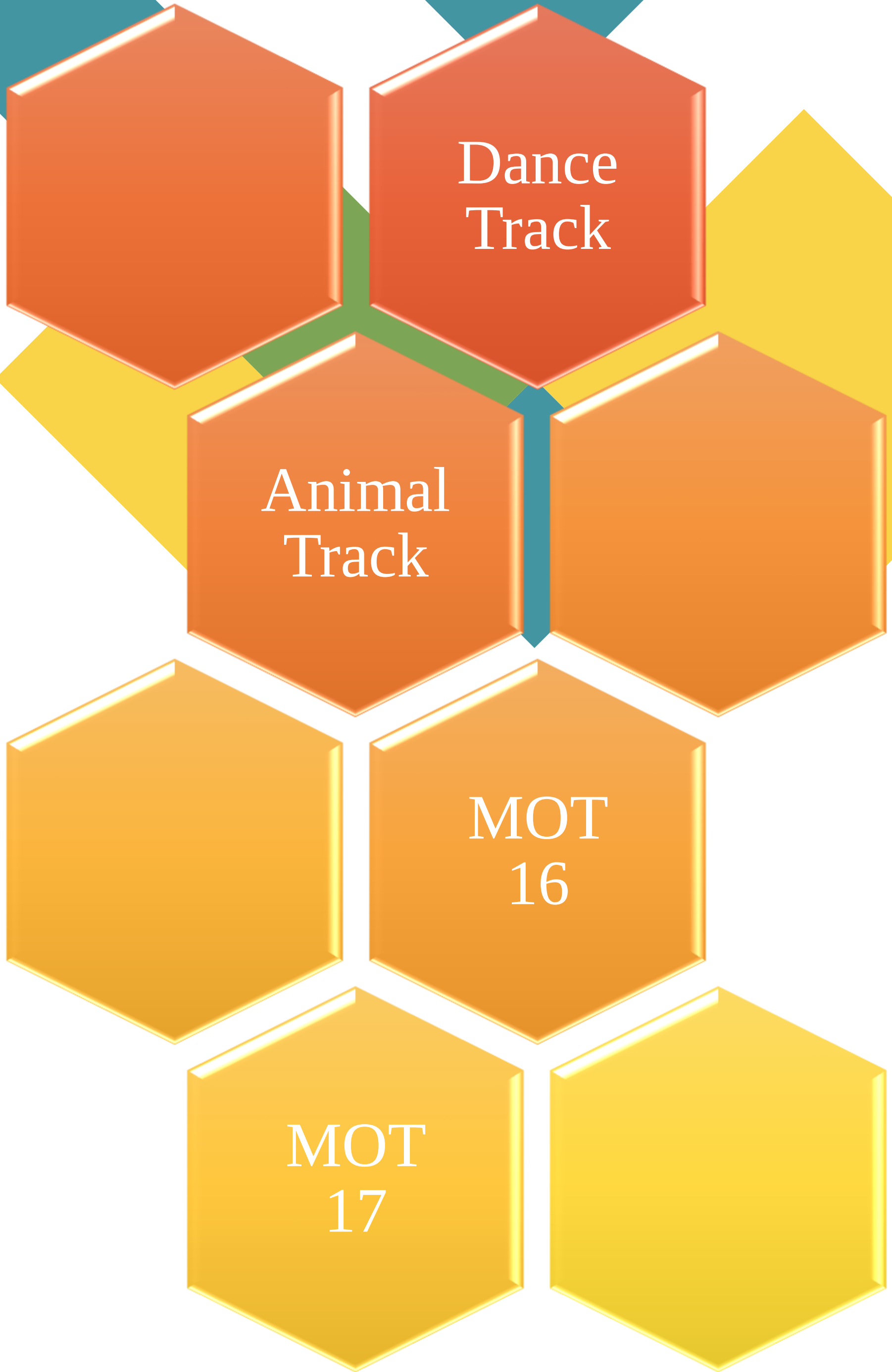
Applying “Quasi-dense Similarity Learning for Multiple Object Tracking method” on DanceTrack

[Link demo](#) [Tracking result](#)

06. Conclusion

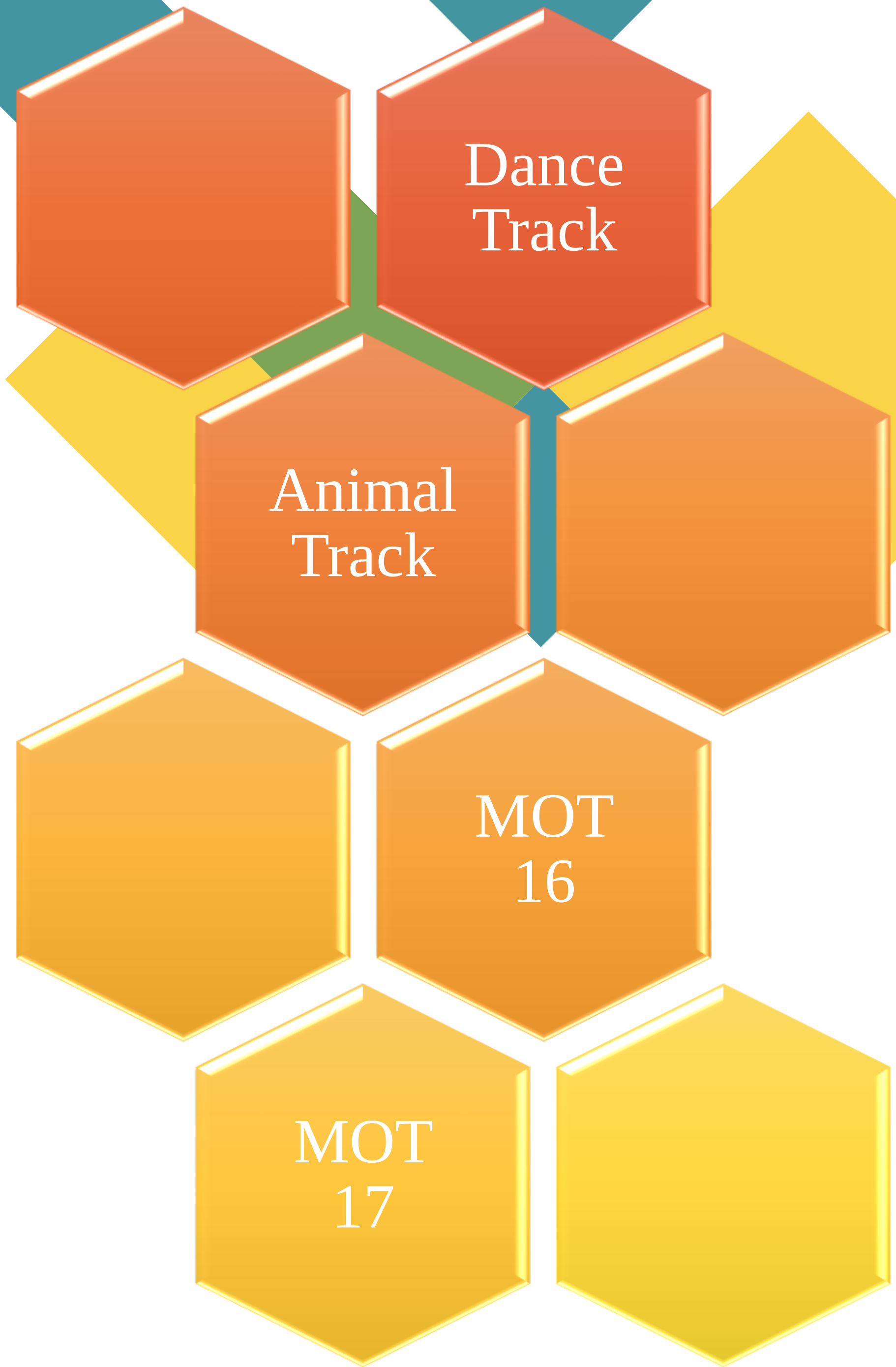
“

QDTrack,
a tracking method based on contrastive learning and quasi-dense matching for instance similarity learning.



“

Quasi-dense is an extremely effective method for tracking objects that <u>have similar appearances</u>



Thank you