Multiobject tracking using deep learning and tracking by detection

Master thesis, academic course 2018-2019

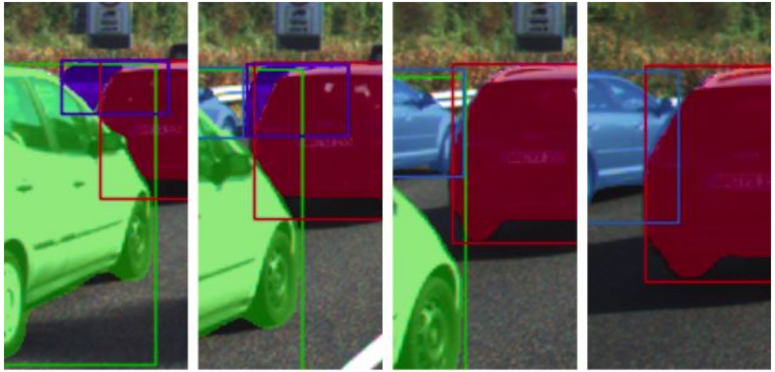
Author: Alexandre Rodríguez Rendo
Tutor: José María Cañas Plaza

URJC
Máster Universitario
en Visión Artificial

Index

1. Introduction

- **Multiple object tracking** in Computer Vision
  - Open problem
  - Multiple applications



1. Introduction

- **Deep learning** in Computer Vision
  - Used in multiple areas surpassing in most cases the results from previous works
  - Also in multiple object tracking

2. Goals

- Build a multi-object tracking application
  - deep learning techniques
  - tracking by detection
  - idea: combine deep learning detections with classic tracking

- Features
  - robust and fast
  - run in resource constrained HW on real-time

- Validate the solution on well-known datasets

3. State of the Art

- **Algorithms' schemes** for object tracking
  - ○ Tracking by detection
  - ○ Tracking, learning and detection
  - ○ Siamese-based tracking
  - ○ Tracking as regression
  - ○ Tracking with RNN

3. State of the Art

- **Datasets** for **multiple** object tracking

MOT16

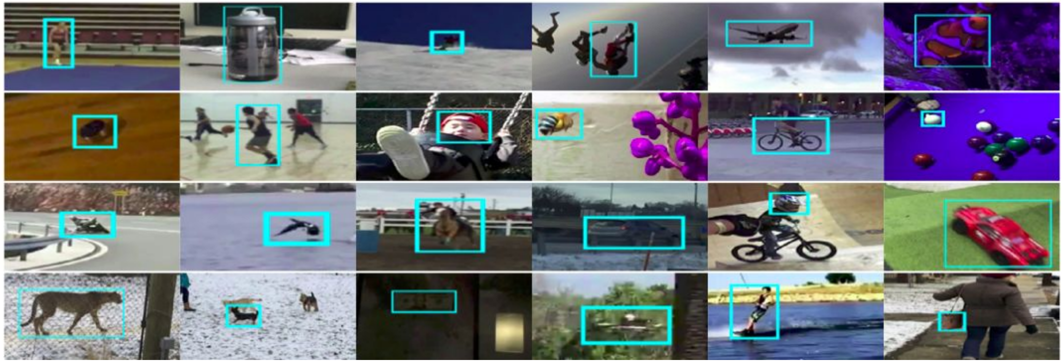
3. State of the Art

- **Datasets** for **multiple** object tracking

PETS

3. State of the Art

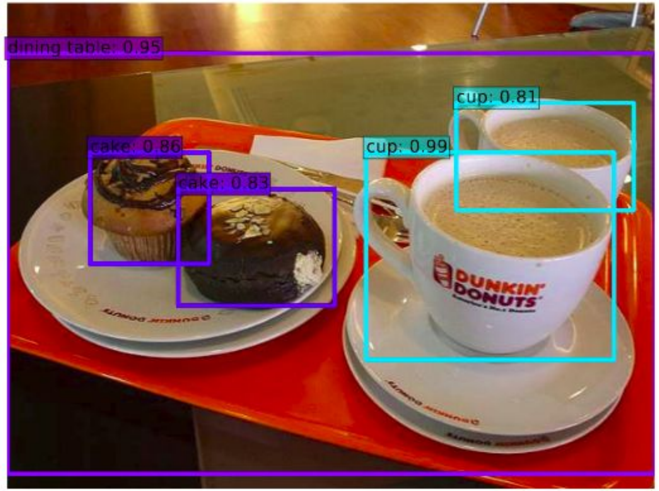
- **Datasets** for **single** object tracking



Need for Speed

3. State of the Art

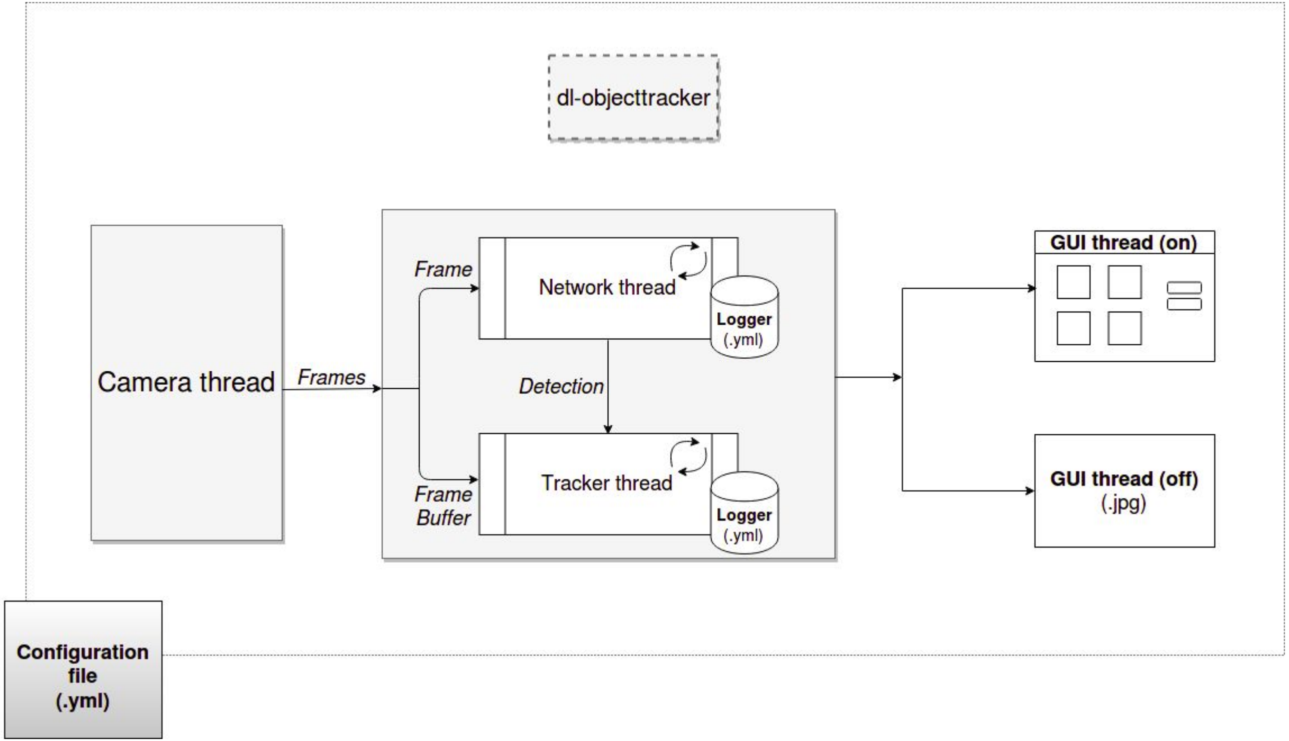
- **Object detection** with **neural networks**
  - Faster R-CNN
  - SSD
  - YOLO
  - Mask R-CNN*



4. dl-objecttracker

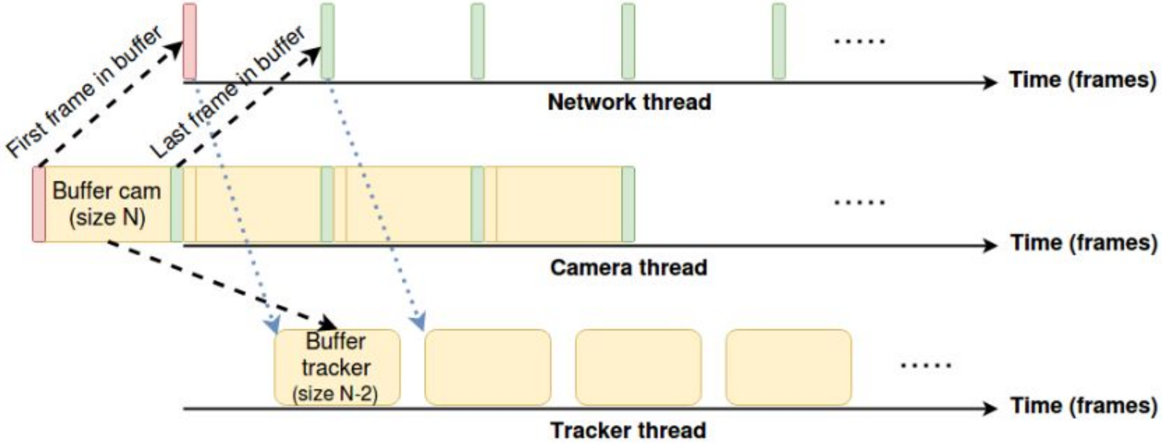
- **Modular** architecture

- python
- configurable
- multiple video sources
- GUI
- logging



4. dl-objecttracker

- How the buffer of frames is handled?



4. dl-objecttracker

- **Neural network** module
  - supports some Keras and Tensorflow models (pretrained)
    - SSD MobileNetV2
    - Faster R-CNN InceptionV2
    - Mask R-CNN InceptionV2
    - SSD VGG
  - provides the object detections to feed the Tracker module

4. dl-objecttracker

- **Tracker** module
  - supports some OpenCV and dlib trackers
    - OpenCV: KCF, BOOSTING, MIL, TLD, MEDIANFLOW, CSRT, MOSSE
    - dlib: CF
  - performs the tracking by detection in a buffer of frames
  - three operating regimes: slow, normal, fast
  - confidence measurement

5. Experiments

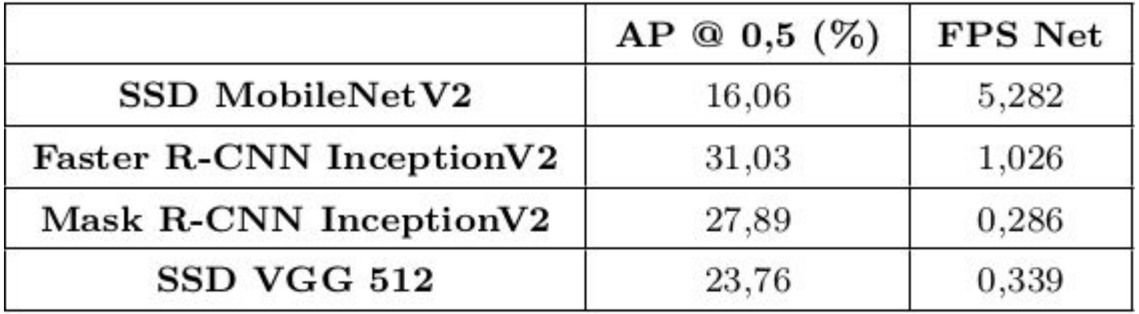
- **Metrics** (Pascal VOC 2010) → *Object Detection Metrics* tool
  - precision
  - recall
  - AP

- **Processing speed** → *dl-objecttracker*

- **Setup**
  - hardware used: CPU (*i7-4510U @ 2.00 GHz x 4*)
  - dataset: *MOT17Det* (train set)

5. Experiments: neural network module

→ Various options available

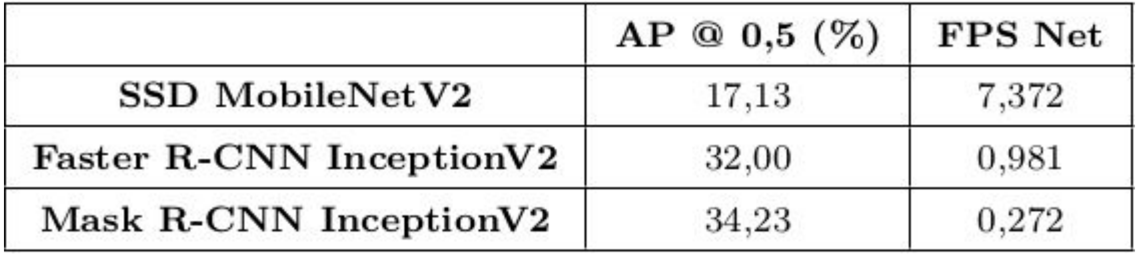


	AP @ 0,5 (%)	FPS Net
SSD MobileNetV2	16,06	5,282
Faster R-CNN InceptionV2	31,03	1,026
Mask R-CNN InceptionV2	27,89	0,286
SSD VGG 512	23,76	0,339

Experiments on MOT17-09 sequence with 512x512 images

5. Experiments: neural network module

→ SSD VGG discarded

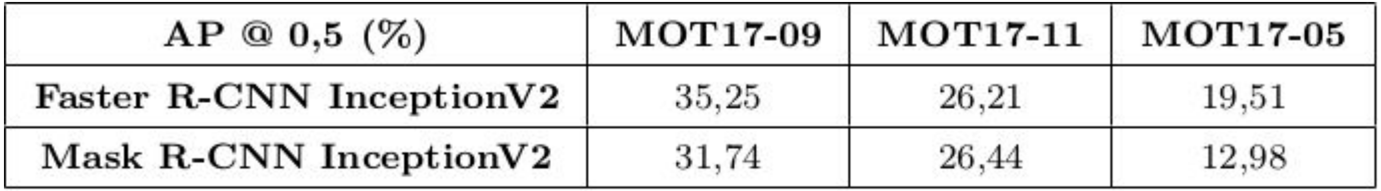


	AP @ 0,5 (%)	FPS Net
SSD MobileNetV2	17,13	7,372
Faster R-CNN InceptionV2	32,00	0,981
Mask R-CNN InceptionV2	34,23	0,272

Experiments on MOT17-09 sequence with 800x800 images

5. Experiments: neural network module

→ Final selection

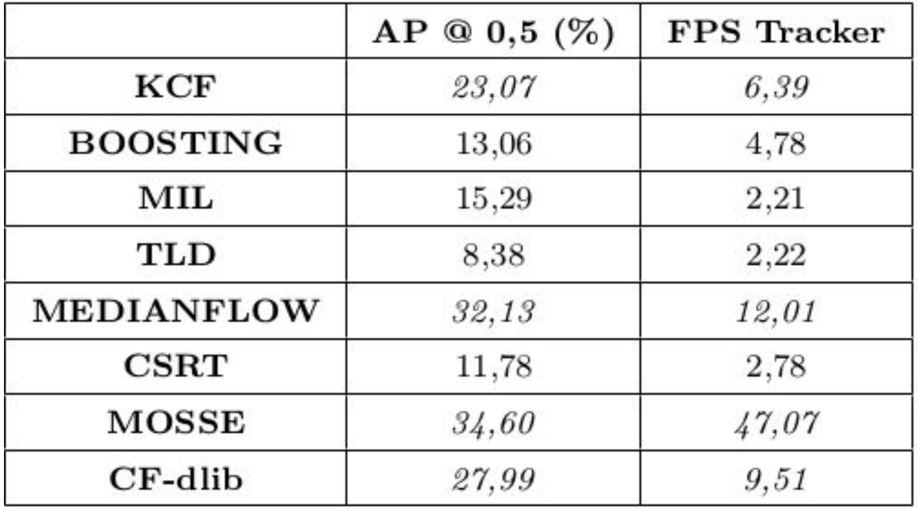


AP @ 0,5 (%)	MOT17-09	MOT17-11	MOT17-05
Faster R-CNN InceptionV2	35,25	26,21	19,51
Mask R-CNN InceptionV2	31,74	26,44	12,98

Experiments on MOT17-09 with 1000x1000 images

5. Experiments: tracker module

→ Multiple tracker options

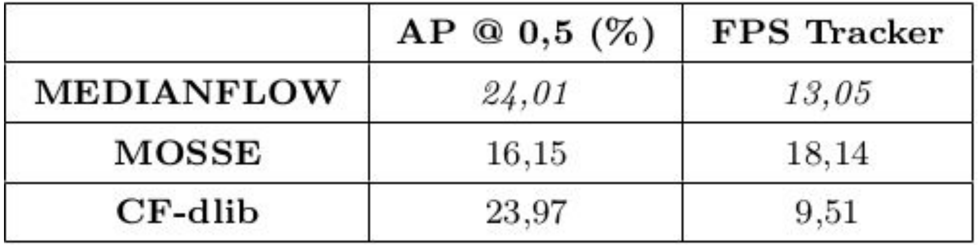


	AP @ 0,5 (%)	**FPS Tracker**
KCF	*23,07*	*6,39*
BOOSTING	13,06	4,78
MIL	15,29	2,21
TLD	8,38	2,22
MEDIANFLOW	*32,13*	*12,01*
CSRT	11,78	2,78
MOSSE	*34,60*	*47,07*
CF-dlib	*27,99*	*9,51*

Experiments on MOT17-09 with 1000x1000 images

5. Experiments: tracker module

→ Best three trackers

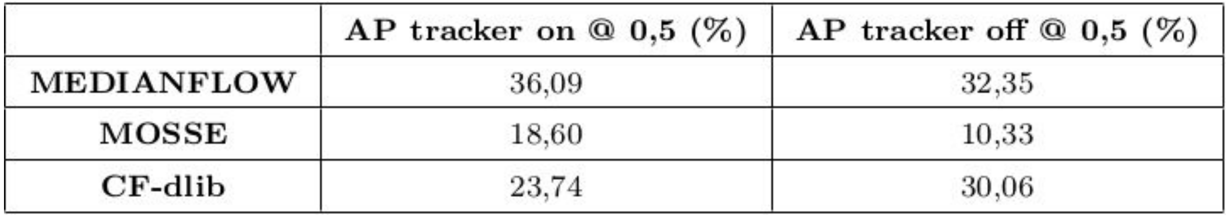


	AP @ 0,5 (%)	FPS Tracker
MEDIANFLOW	*24,01*	*13,05*
MOSSE	16,15	18,14
CF-dlib	23,97	9,51

Experiments on MOT17-05 with 1000x1000 images

5. Experiments: tracker module

→ Confidence influence



	AP tracker on @ 0,5 (%)	AP tracker off @ 0,5 (%)
MEDIANFLOW	36,09	32,35
MOSSE	18,60	10,33
CF-dlib	23,74	30,06

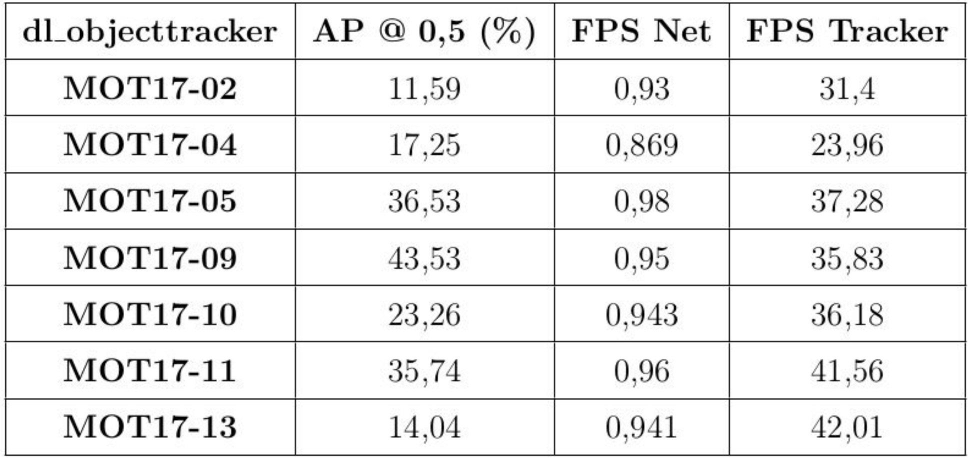
Confidence influence on tracking performance on MOT17-05

5. Experiments: final solution

- Neural network
  - Faster R-CNN InceptionV2
  - input size 400x400
  - confidence threshold 0,5

- Tracker
  - MedianFlow
  - using tracker confidence

5. Experiments: final results



dl_objecttracker	AP @ 0,5 (%)	FPS Net	FPS Tracker
MOT17-02	11,59	0,93	31,4
MOT17-04	17,25	0,869	23,96
MOT17-05	36,53	0,98	37,28
MOT17-09	43,53	0,95	35,83
MOT17-10	23,26	0,943	36,18
MOT17-11	35,74	0,96	41,56
MOT17-13	14,04	0,941	42,01

MOT17Det train set

6. Conclusions

1. Region-based object detection neural networks obtain the best accuracy
2. MedianFlow seems to be the best tracker available in OpenCV
3. Confidence is useful to discard bad tracking performance in OpenCV, not occurs the same in dlib
4. Image input size is key when working in limited hardware
5. Final solution performs best on lowly crowded sequences

6. Conclusions: future works

1. Train neural network models on MOT datasets
2. Use dlib multiprocessing in tracking
3. Obtain the best configuration in a different way
4. Improve the metrics calculation → IDs → MOTA, MOTP, ...
5. Test the application in other non-GPU devices and with GPU acceleration
6. Try weights quantization techniques

Thank you for your attention

Annex: links

- MOT17Det results
  https://motchallenge.net/results/MOT17Det/

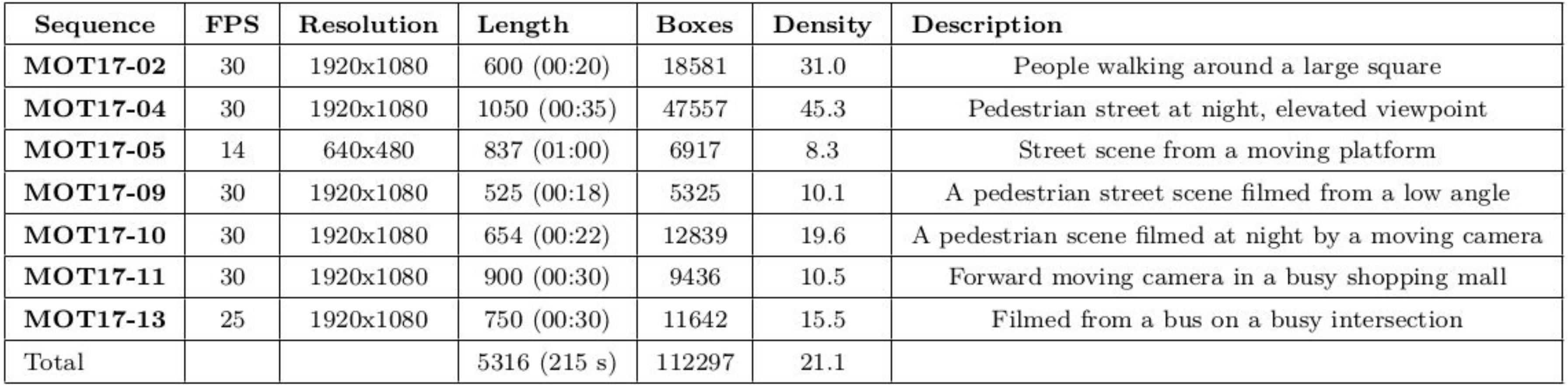
- MOT CVPR 2019 tracking results
  https://motchallenge.net/results/CVPR_2019_Tracking_Challenge/

- Object Detection Metrics
  https://github.com/rafaelpadilla/Object-Detection-Metrics

- MedianFlow paper https://ieeexplore.ieee.org/abstract/document/5596017

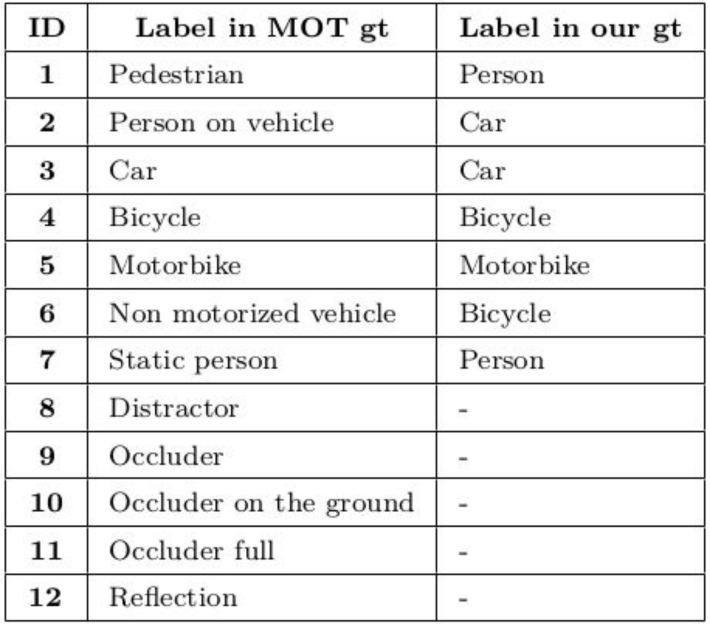
- Demo results https://www.youtube.com/watch?v=LyN2aeIFFHI

Annex: dataset description



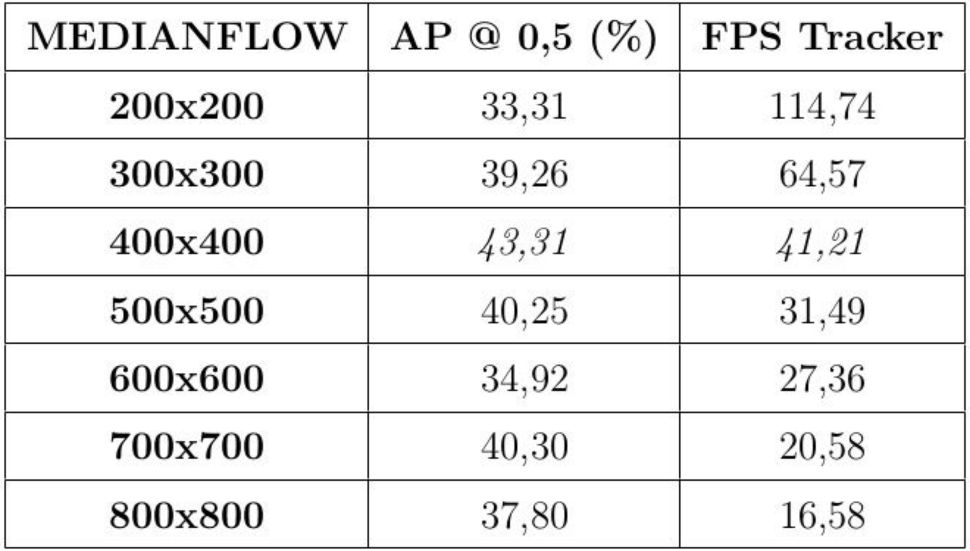
Sequence	FPS	Resolution	Length	Boxes	Density	Description
MOT17-02	30	1920x1080	600 (00:20)	18581	31.0	People walking around a large square
MOT17-04	30	1920x1080	1050 (00:35)	47557	45.3	Pedestrian street at night, elevated viewpoint
MOT17-05	14	640x480	837 (01:00)	6917	8.3	Street scene from a moving platform
MOT17-09	30	1920x1080	525 (00:18)	5325	10.1	A pedestrian street scene filmed from a low angle
MOT17-10	30	1920x1080	654 (00:22)	12839	19.6	A pedestrian scene filmed at night by a moving camera
MOT17-11	30	1920x1080	900 (00:30)	9436	10.5	Forward moving camera in a busy shopping mall
MOT17-13	25	1920x1080	750 (00:30)	11642	15.5	Filmed from a bus on a busy intersection
Total			5316 (215 s)	112297	21.1	

Annex: dataset ground truth



ID	Label in MOT gt	Label in our gt
1	Pedestrian	Person
2	Person on vehicle	Car
3	Car	Car
4	Bicycle	Bicycle
5	Motorbike	Motorbike
6	Non motorized vehicle	Bicycle
7	Static person	Person
8	Distractor	-
9	Occluder	-
10	Occluder on the ground	-
11	Occluder full	-
12	Reflection	-

Annex: image size



MEDIANFLOW	**AP @ 0,5 (%)**	**FPS Tracker**
200x200	33,31	114,74
300x300	39,26	64,57
400x400	*43,31*	*41,21*
500x500	40,25	31,49
600x600	34,92	27,36
700x700	40,30	20,58
800x800	37,80	16,58

Image input size experiments on MOT17-09