Monitoring Traffic Flows via Unsupervised Domain Adaptation

Luca Ciampi, Claudio Gennaro, Giuseppe Amato

luca.ciampi@isti.cnr.it

http://aimir.isti.cnr.it

23 September 2020

I-CiTies 2020

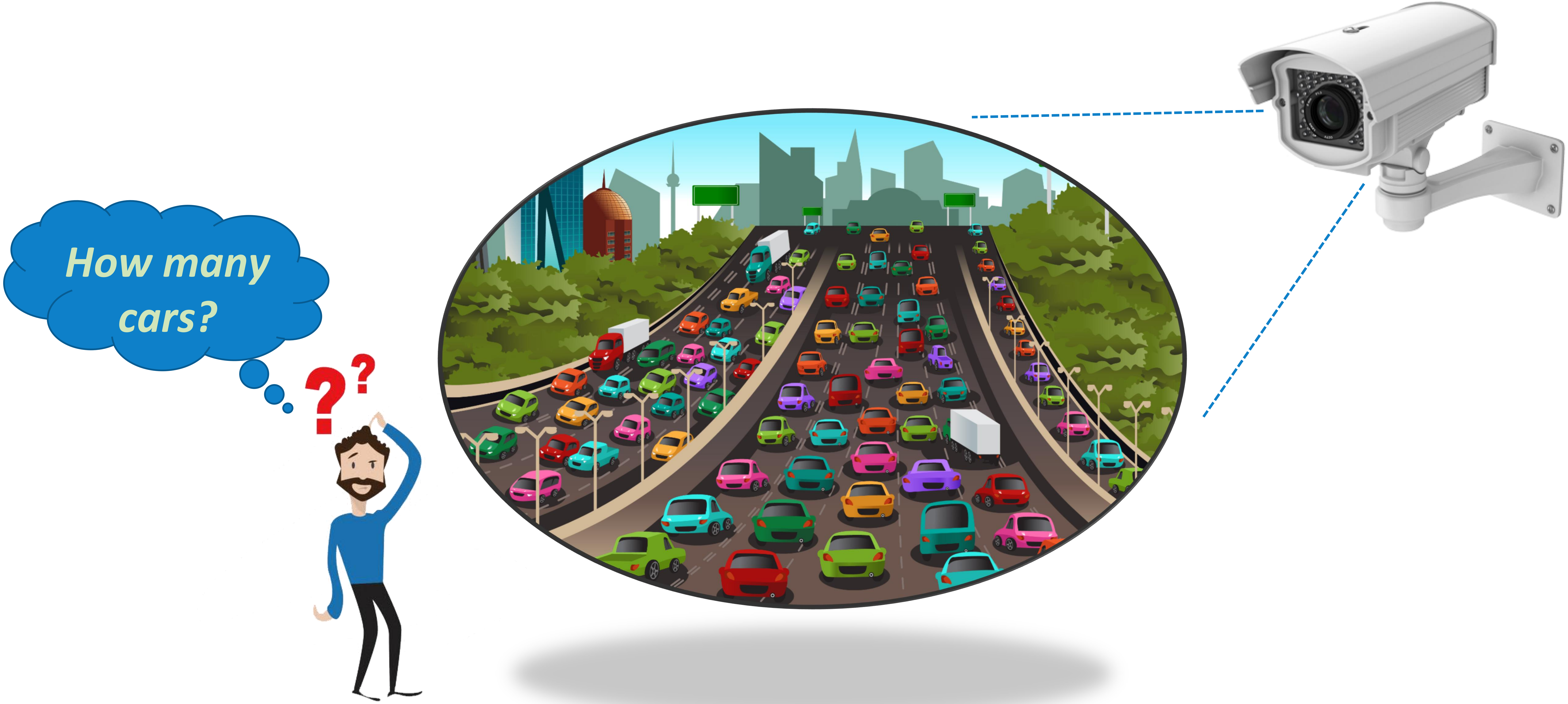
Outline

THE PROBLEM


CRUCIAL TO IMPROVE URBAN ENVIRONMENT AND LIFE OF CITIZENS

- ➡ ***CITY MOBILITY***
- ➡ ***POLLUTION MONITORING***
- ➡ ***INFRASTRUCTURE MANAGEMENT***


VISUAL COUNTING AND DENSITY ESTIMATION OF TRAFFIC FLOWS

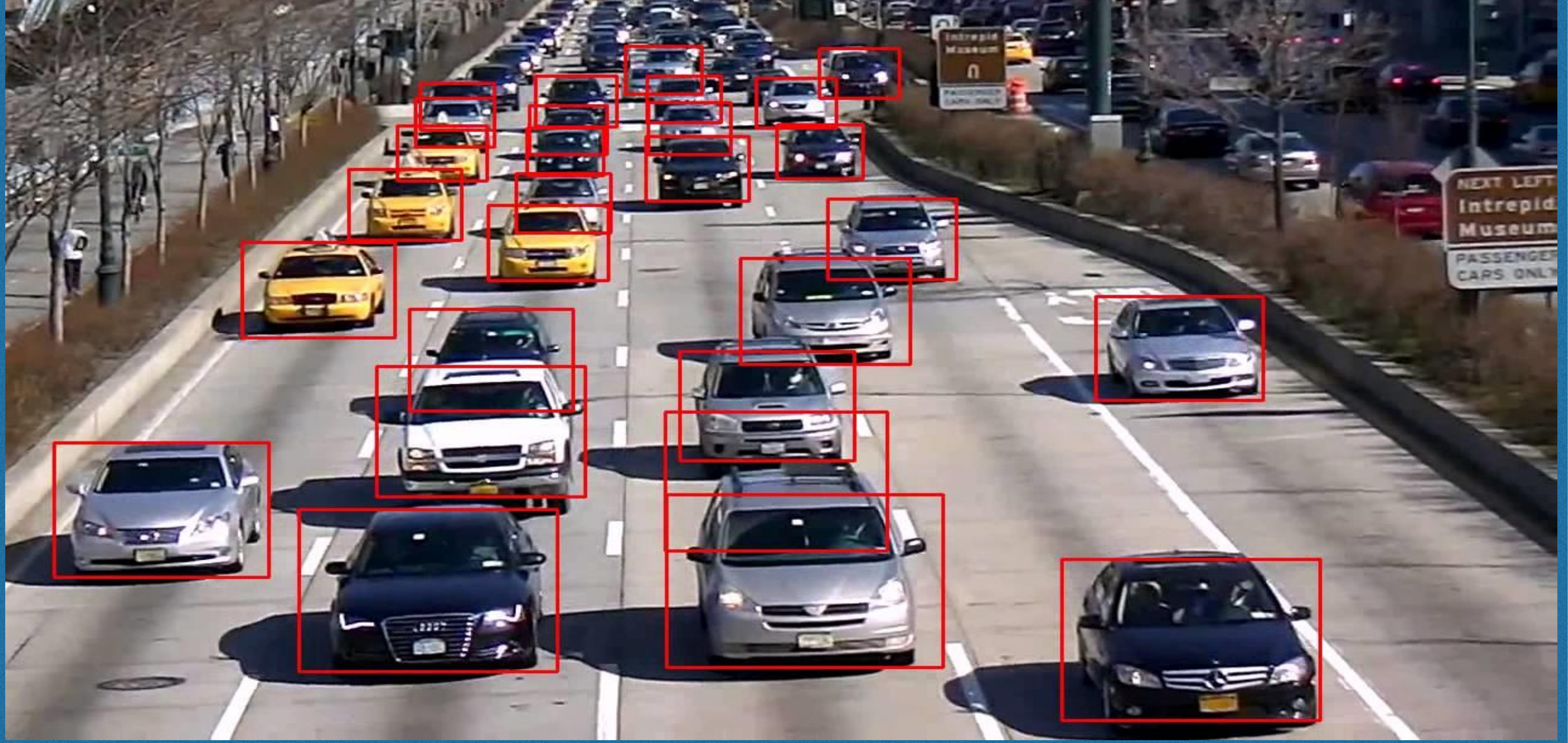


Current Approaches

Detection Based

- Supervised Technique
- Localize instances and count them

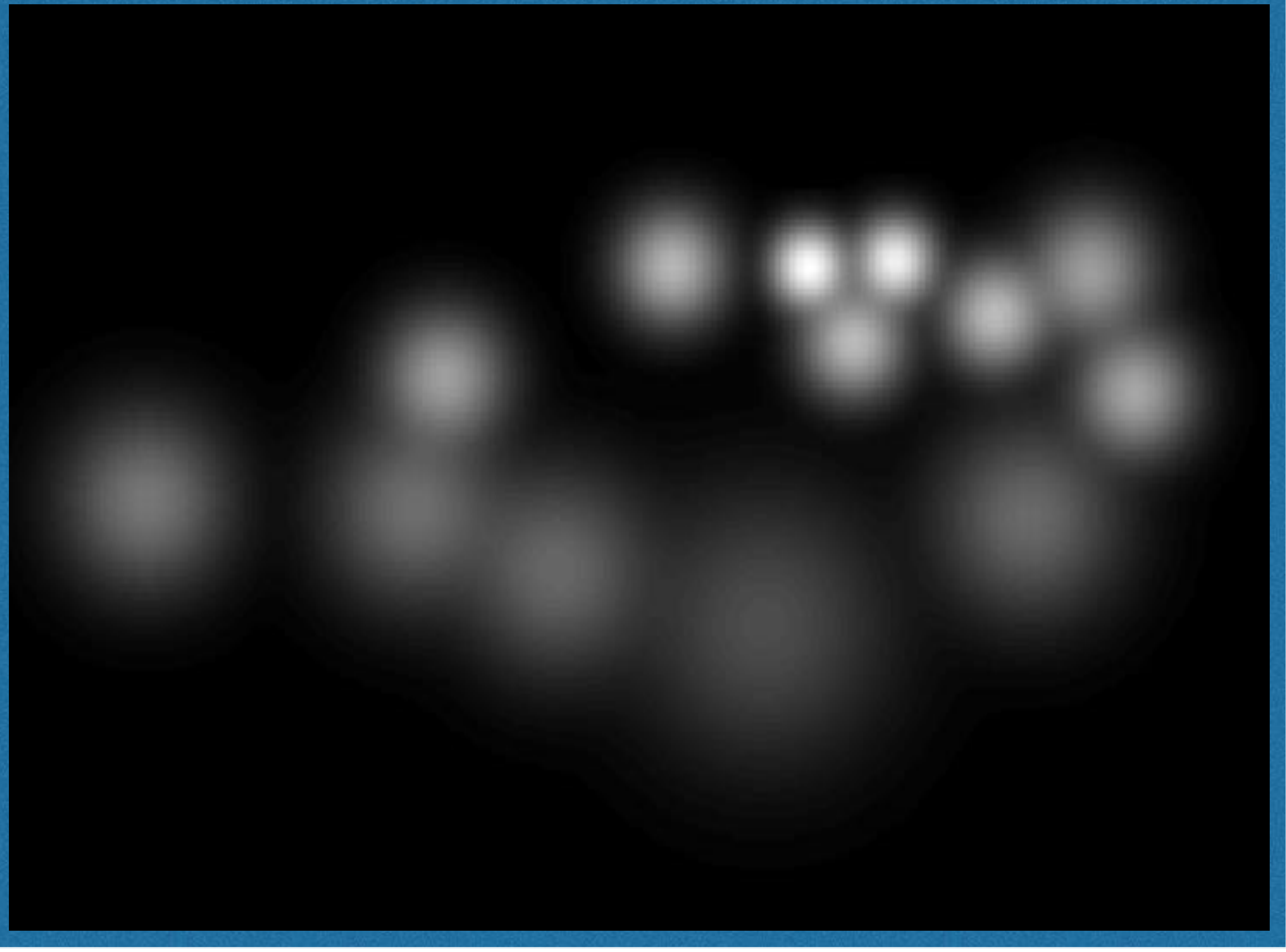
Not feasible in every scenario!



Regression Based

- Supervised Technique
- Regression from image features to total number or to density map (that is then integrated)

It works in very crowded scenario!



DOMAIN SHIFT PROBLEM

- A massive amount of <u>labeled</u> data is needed to train these algorithms
- In many real-world applications there is a large Domain Shift between the distribution of the train (source) and test (target) domains
  - Significant drop in performance at inference time
- Different smart cameras across the city are subject to various visual conditions (luminance, position, context)
  - Different performances for each of them
  - Unfeasible to collect and label data for every different scenarios
  - Difficult to effectively scale-up the system as new cameras are added

Ground Truth Generation

- Gaussian over each object
- Spread estimated with some heuristic
- Background to zero
- Summing up pixel values → number of objects

⇨ **Lot of human effort**

⇨ **Time Consuming**

⇨ **Just an approximation**

PROPOSED SOLUTION

UNSUPERVISED DOMAIN ADAPTATION (UDA)

- ⇨ UDA → class of techniques that aims to mitigate the Domain Shift problem without the need of labeled data in the target domain

- ⇨ CNN-based UDA algorithm for traffic density estimation and counting
  - ⇨ Adversarial Learning in the output space

- ⇨ Experiments considering different types of Domain Shift:
  - ⇨ *Camera2Camera* Domain Shift → different cameras in train and test phases
  - ⇨ *Day2Night* Domain Shift → day images for training and night images for test
  - ⇨ *Synthetic2Real* Domain Shift → synthetic images for training and real images for test

The WebCamT Dataset

- 5,000 images belonging to 10 different cameras of urban scenarios
- Low-resolution, large perspectives, heavily occluded
- Manual annotated with bounding boxes
- Camera2Camera Domain Shift → 7 cameras for training, 3 for testing

The Night and Day Instance Segmented Park Dataset (NDISPark)

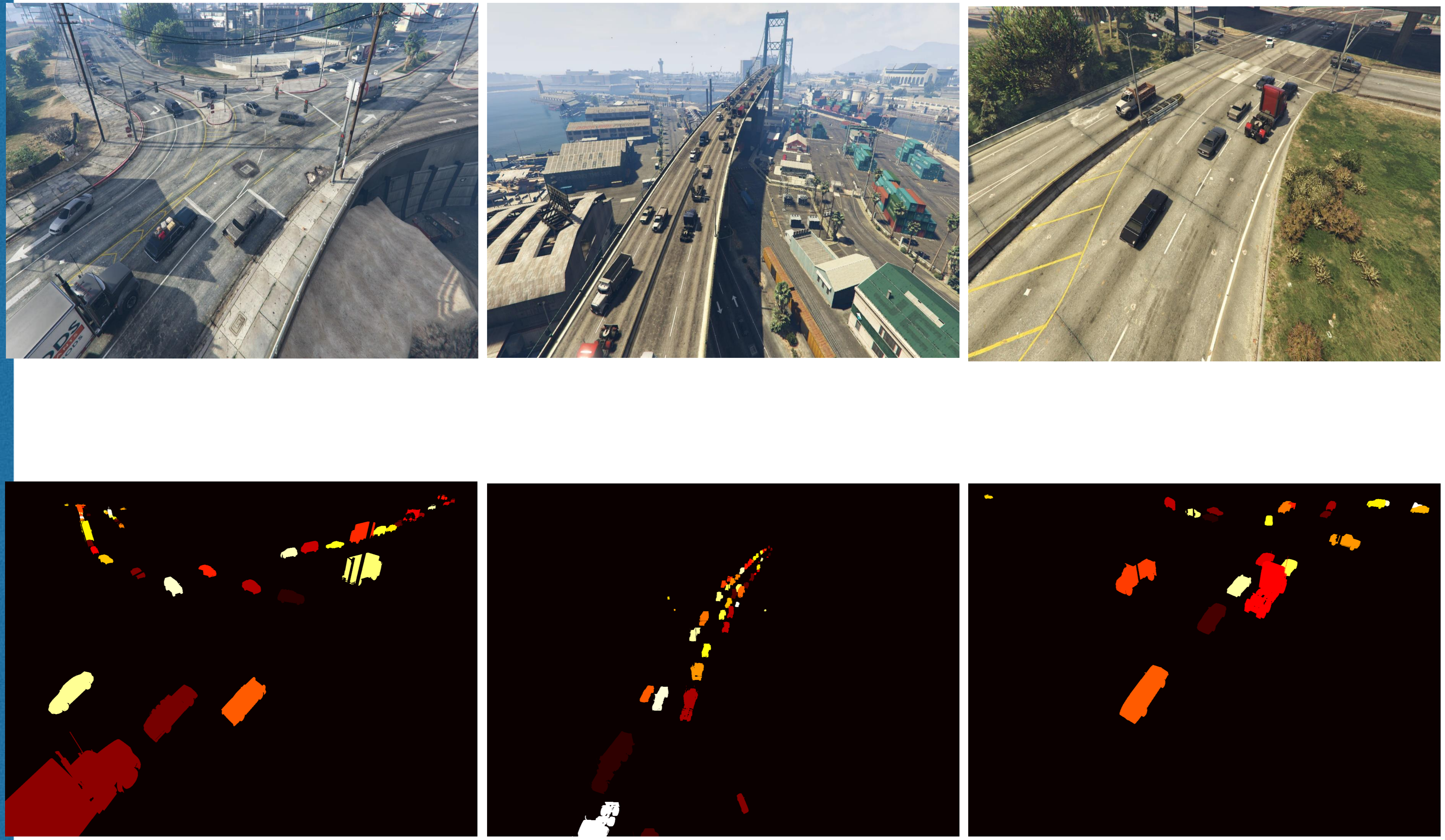
- ~250 images of cars in parking lots
- Manually annotated with instance segmentation labels → Accurate density maps
- Manual annotated with bounding boxes
- Images taken during the day and the night, showing utterly different lighting conditions → Day2Night Domain Shift

Gathering labeled training data from virtual worlds

Grand Traffic Auto Dataset

- ~15.000 high congested traffic scenes
- Collected from Grand Theft Auto videogame → Synthetic2Real Domain Shift
- Many different perspectives, illumination, contexts
- <u>Automatically</u> annotated
- Per-pixel labels → Accurate Density Maps

For a total of ...



automatically labelled vehicles in urban scenarios (after some cleanup)

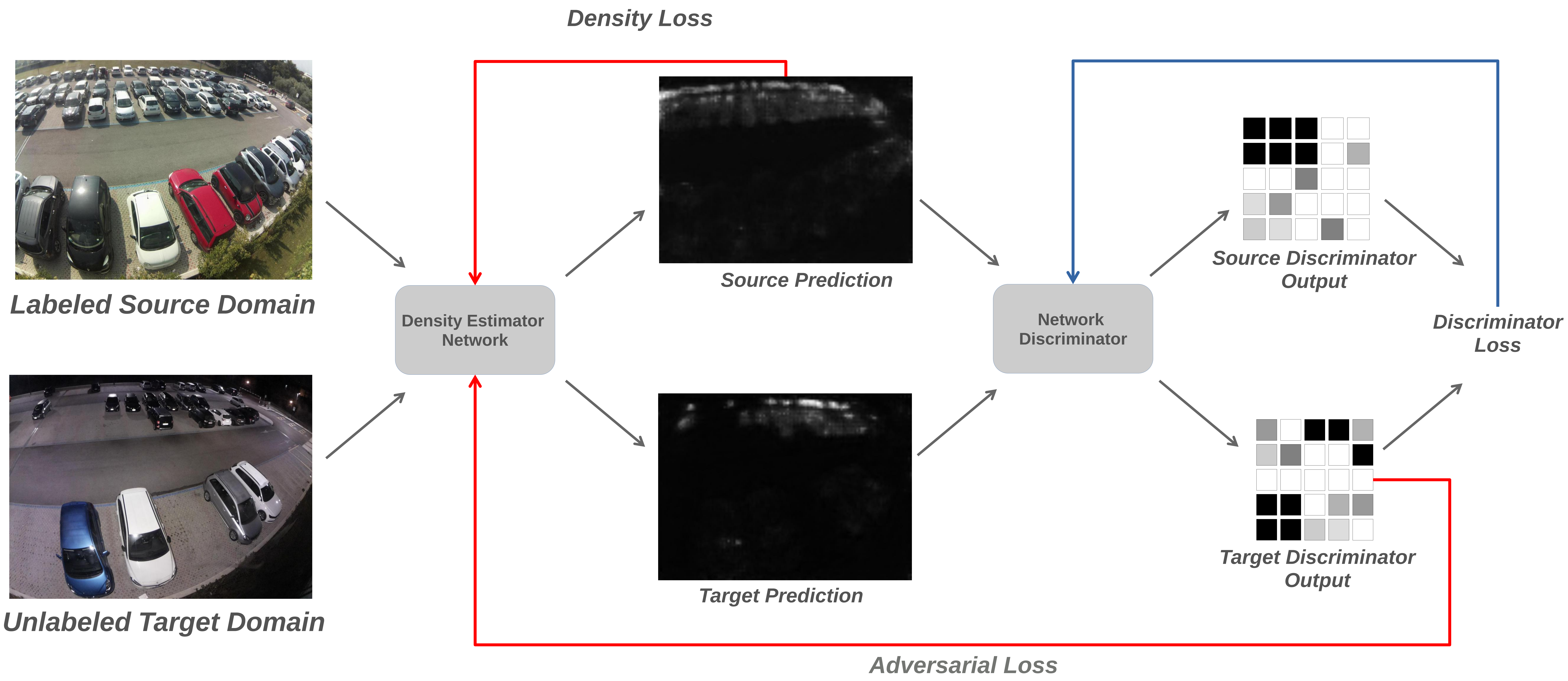
Considered Domain Shifts - Recap

Camera2Camera

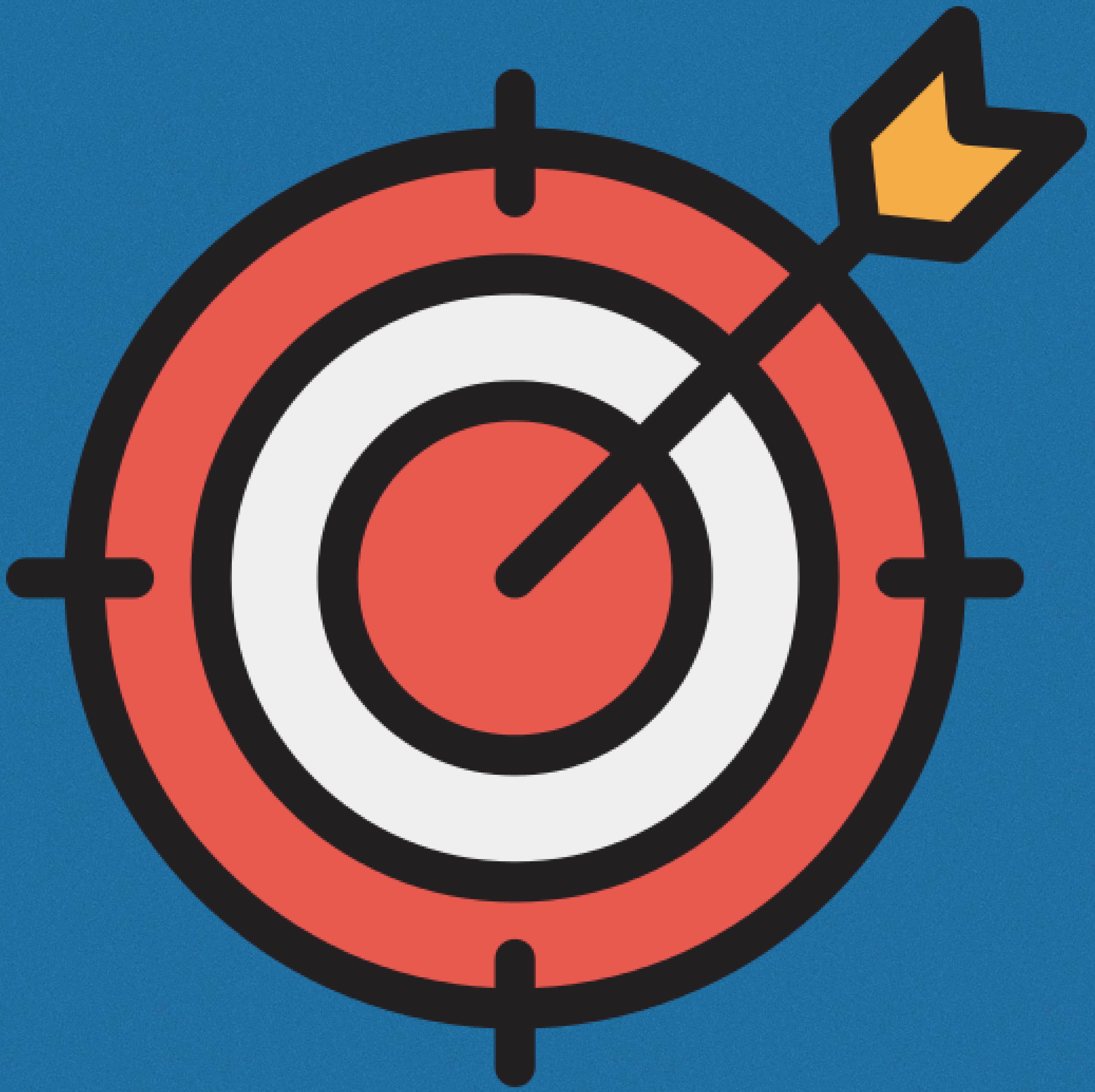
Day2Night

Synthetic2Real

The Architecture



PRELIMINARY RESULTS



Metrics

➡ MAE: Mean Absolute Error

$$\frac{1}{N}\sum_{n=1}^{N}|c_n^{gt} - c_n^{pred}|$$

➡ RMSE: Root Mean Squared Error

$$\sqrt{\frac{1}{N}\sum_{n=1}^{N}(c_n^{gt} - c_n^{pred})^2}$$

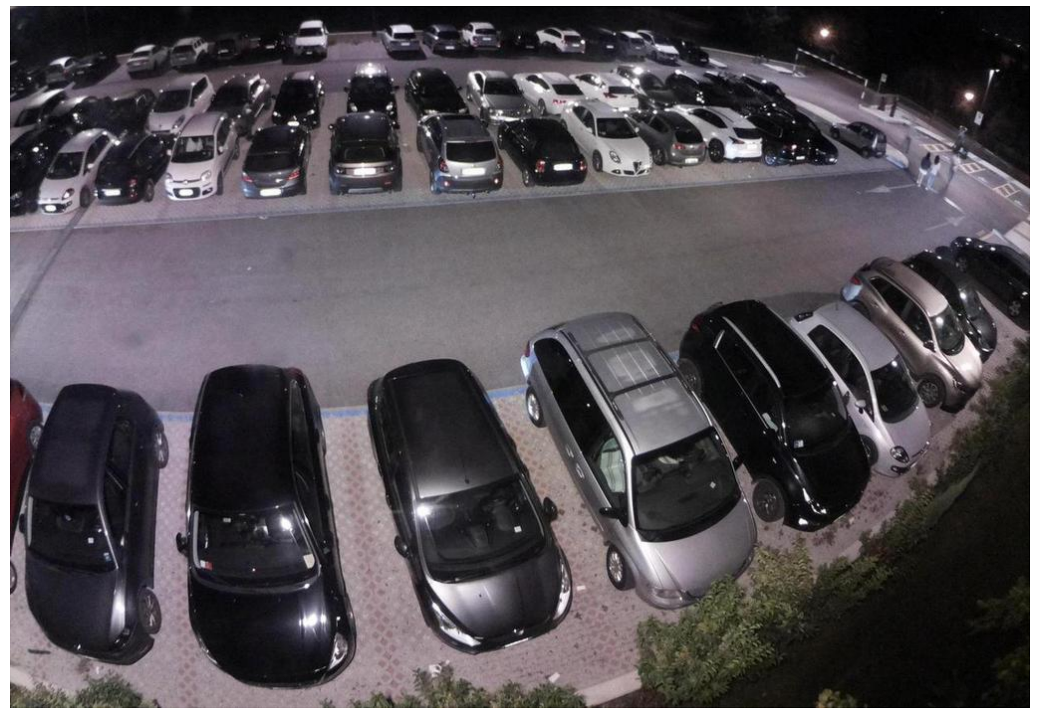
➡ ARE: Average Relative Error

MAE / num_cars

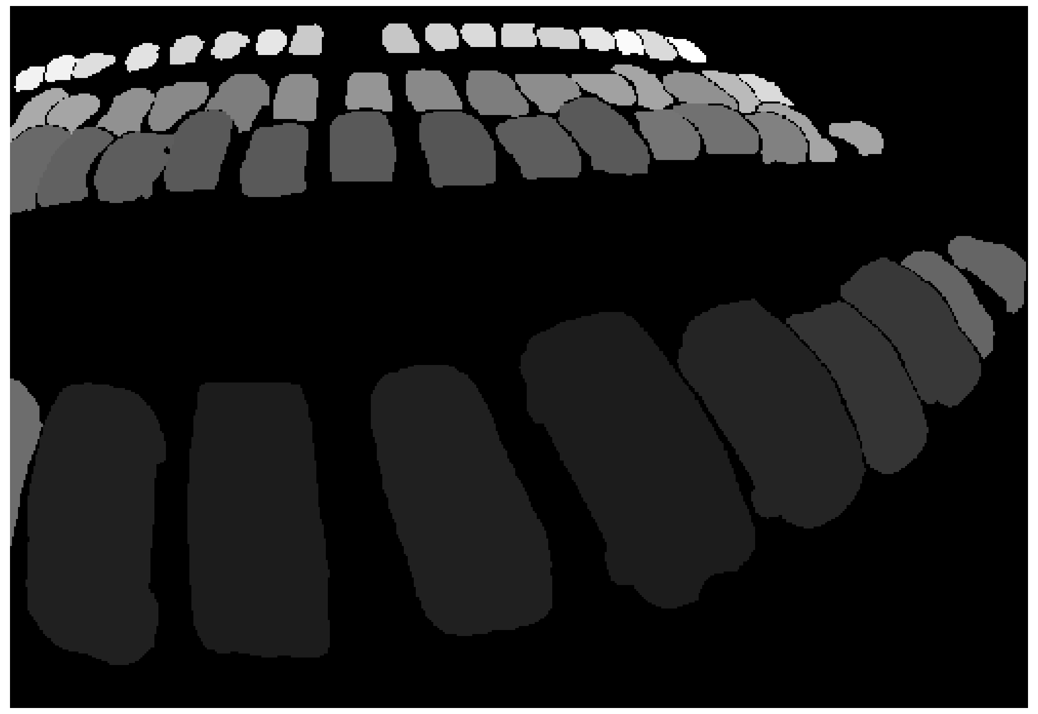
Results

	Camera2Camera Domain Shift	Day2Night Domain Shift	Synthetic2Real Domain Shift
Baseline (without Discriminator)	3,24	1,70	4,10
Our Method (with Discriminator)	2,86	1,45	3,88

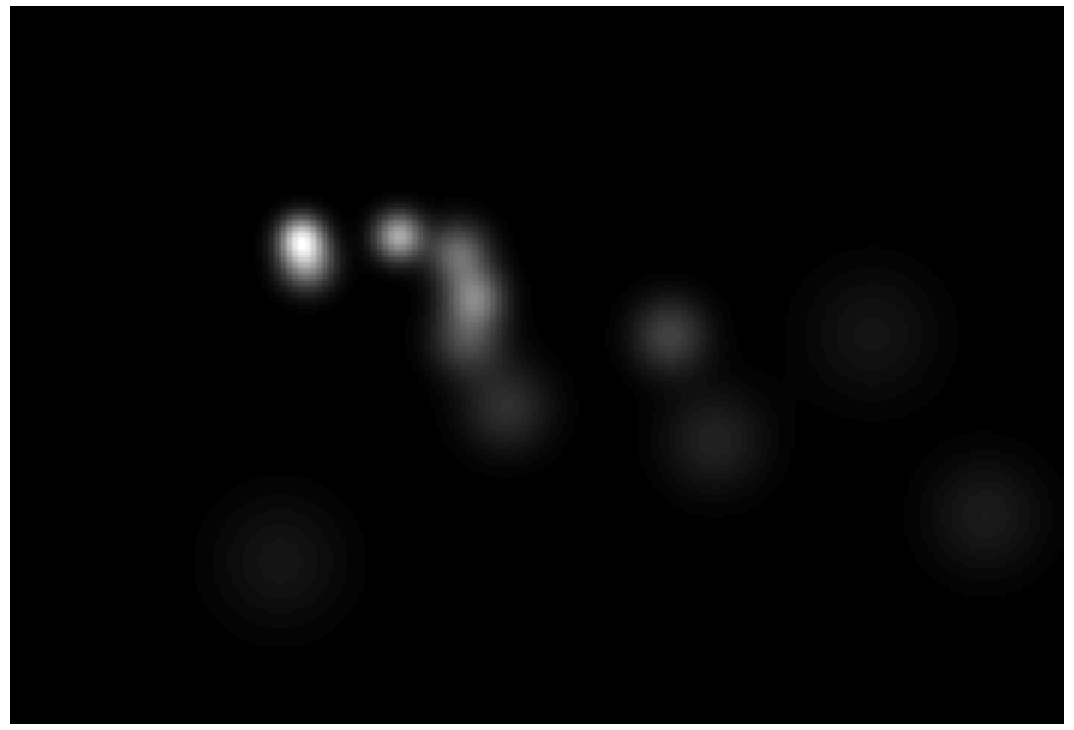
Output Predictions - Examples



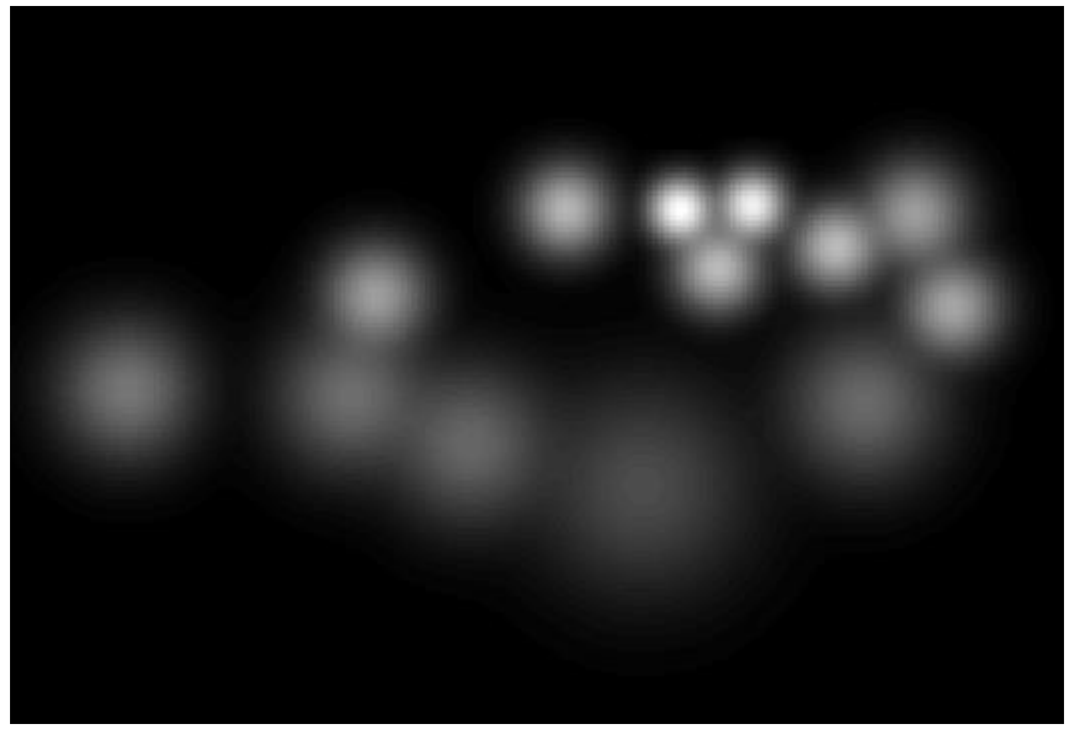
GT Count: 56



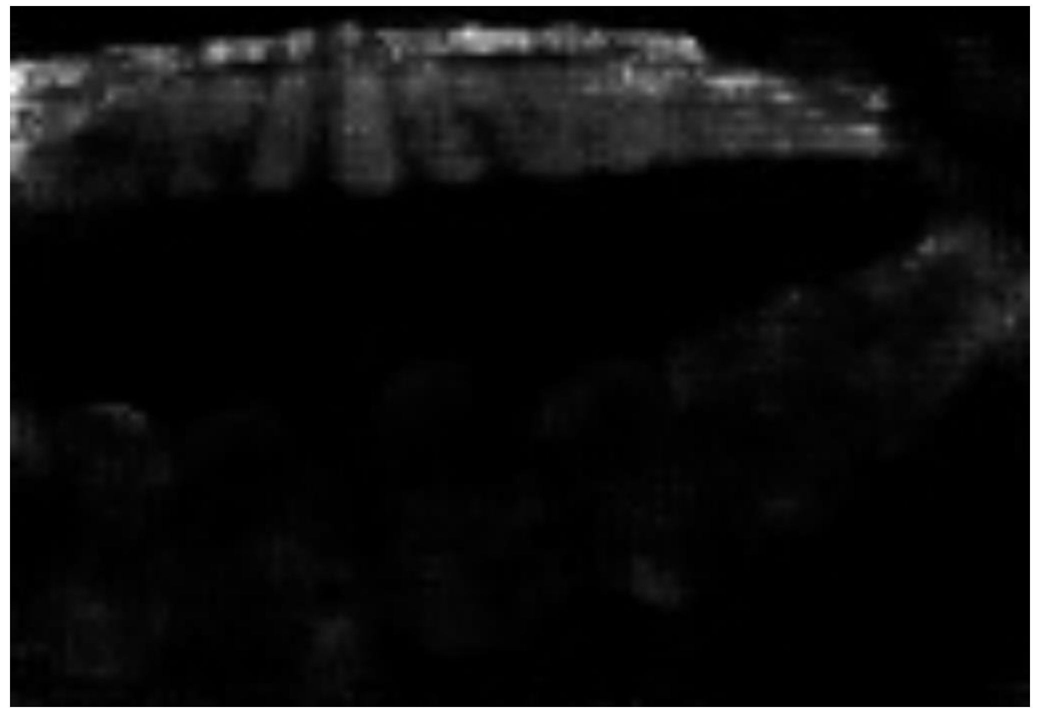
GT Count: 12



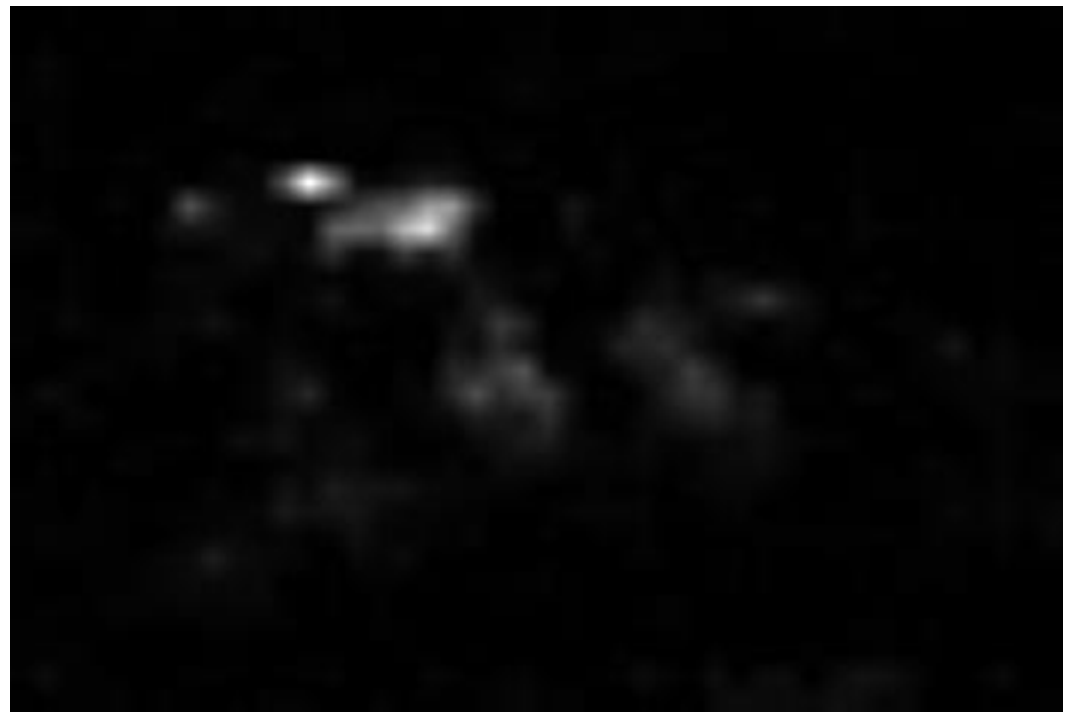
GT Count: 13



Pred Count: 53



Pred Count: 11



Pred Count: 14

CONCLUSIONS

⇨ Building on a CNN-based density estimator, the proposed methodology can generalize to new sources of data for which there are no annotations available

⇨ We achieve this generalization by adversarial learning, whereby a discriminator attached to the output forces similar density distribution in the target and source domains

⇨ Experiments show a significant improvement relative to the performance of the model without domain adaptation

Thanks for your attention!

Questions?

Luca Ciampi <luca.ciampi@isti.cnr.it>