



Smart City Solution: AI-based Analysis of Public Parking Spaces

Team:

Basak Erdamar, Katarina Golubovic, Ashish Kumar, Niels Schlegel, Nastassia Vasileuskaya

Mentors: Tanja Pfaffel, Thorsten Philipp **Co-Mentor:** Msc Philippe Sünnen

Project Leader: Dr. Ricardo Acevedo Cabra **Supervisor:** Prof. Dr. Massimo Fornasier

Meet our team!





Motivation

Data

Algorithm

Transfer Learning

Parked Cars Analysis

Mobility Analysis

Web Application Results Outlook

Motivation

- Bikers are on the same road as cars, trucks and buses
- Too many traffic participations

- Infrastructure not well organized
- Traffic flow is not fluent

- No insight in parking spaces
- Too many cars in the center



Goal

Prioritization

Become a smart city as a part of project "Smart City Model Projects: Urban Development and Digitalization"

Final vision: Having a slower but fluent traffic flow without disturbances



Data



18. February 2020 - 05. March 2020 24 hours/day **4595** videos

Low quality videos

640x360 resolution

8,7 GB

5 frames/second

High quality videos

3840x2160 resolution

~**800** GB

25 frames/second

Data







18. February 2020 - 05. March 2020 24 hours/day **4595** videos 24. February 2020 - 29. February 2020 Mon, Tue, Wed, Thu, Sat 07:00-09:00; 12:00-14:00; 16:00-18:00;

Low quality videos

640x360 resolution

8,7 GB

5 frames/second

High quality videos

3840x2160 resolution

~**800** GB

25 frames/second

Compressed videos

960x540 resolution

~**11.3** GB

25 frames/second



[class probability (p_c), bounding box coordinates (b_x , b_y : midpoint, b_h : height, b_w : width), detected class (c)]

Retraining

MOTIVATION

- Challenging camera angle
- Many false negative detection in the area far away from the camera
- Overlapping objects

SET UP

- Google Colab GPU (Tesla T4)
- No. of iterations = No. of classes x 2000
- Training time 10 25 hours (depending on the no. of classes and given GPU power)

LABELLING DATA

PRE-TRAINED WEIGHTS OF CONVOLUTIONAL LAYERS + LAST FULLY CONNECTED LAYER - RETRAIN WEIGHTS



DETECTION OF PARKED CARS



Analysis of parking spaces

GOAL:

- Define the parking area
- Count the number of free and occupied spaces
- Prepare the output for analysis
- Visualize occupation of parking spaces

Data challenges



Methodology



COMBINE BOTH

Parking Area

- Customer defined parking spaces area = region of interest (ROI)
- Parking spaces on the right side priority
- ROI is defined with coordinates of polygon that is sent as an input to a model



Transfer Learning | Before

- Cars detection using YOLOv3 weights before transfer learning
- **Before:** 9/11 detected



Transfer Learning | After

- 209 pictures used for labeling
- Critical situations with overlapping
- Training time: 20-25h
- After: 11/11 detected



Detection only on the parking area

Cars detection on the ROI

- Using YOLOv3 and retrained weights on the defined parking spaces area
- Parked car = Center of bounding box in defined ROI



-Tracking and Detection

Tracking of parking process

- Tracking the cars
- Defined parking lines
- Counting parked cars based on the intersection of the custom-defined parking lines and lines of bounding boxes of cars



https://www.youtube.com/watch?v=33RsLEN7FtE&feature=youtu.be

Comparison of Accuracy

 $Accuracy = \frac{No \text{ of correctly detected cars on the parking space}}{No \text{ of ground-truth cars on the parking space}}$

Video	Low quality %	Compressed (Higher) quality %
2020-02-24 07:13:00	100	100
2020-02-24 08:05:40	100	100
2020-02-24 12:05:30	100	100
2020-02-24 13:06:40	87,5	87,5
2020-02-24 16:05:10	81,3	93,8
2020-02-24 17:11:25	88,9	100

Output and Outlook

OUTPUT:

• Dataset used for the future analysis

parking_spaces.head()

	Date_time	free_parking_spaces	occupied_spaces
0	2020-02-26 13:01:22	8	10
1	2020-02-26 13:03:06	8	10
2	2020-02-26 13:06:28	8	10
3	2020-02-26 13:08:12	8	10
4	2020-02-26 13:11:34	9	9

OUTLOOK:

- Running data for longer time or on higher computer power
- Analyse the parking spaces on the different times of the year or during holiday time.
- Change the angle of the camera to achieve the best accuracy.

TRAFFIC AND MOBILITY ANALYSIS



- 3 Tasks of Mobility Analysis



VEHICLE COUNTING

SPEED ESTIMATION

Heatmap – Detection and Marking of Objects

• Goal

→ Visualize the moving behavior of pedestrians and bicyclists

- Approach
 - 1. Detect objects at every nth frame
 - 2. Draw point in the middle of the detection in a class-dependent color
 - 3. Compare supposed & actual tracks



Object Detection – Before & After Retraining

Before



AP-bicyclists: 0% AP-pedestrians: 15.90% After



AP-bicyclists: 47.29% AP-pedestrians: 25.15%

Heatmap - Output

Output-Table (example):

Time	x-val	y-val	class
2020-02-24 09:15:04	0.34	0.45	pedestrian
2020-02-24 09:15:05	0.35	0.47	pedestrian
2020-02-24 09:15:05	0.2	0.8	bicyclist



https://youtu.be/2VcakEroftE

Outlook

- Train on 6 classes
- Direction detection
- Narrowing zone detection

⁻ 3 Tasks of Mobility Analysis



Vehicle Counting

- Approach
 - 1. Detection-YOLOv3



Vehicle Counting

- Approach
 - 1. Detection- YOLOv3

2. Tracking- SORT

2.1 IOU of 2 consecutiveframes2.1 Index IDs attached andtracked



SORT: Simple Online and Real-Time Tracking IOU: Intersection Over Union

Vehicle Counting

- Approach
 - 1. Detection- YOLOv3
 - 2. Tracking- SORT 2.1 IOU of 2 consecutive frames 2.1 Index IDs attached and tracked
 - 3. Counter- Line intersection w.r.t old and new midpoint coordinates



https://www.youtube.com/watch?v=3K14KYX2Wa8

Vehicle Counting - Performance

Observations:

- Trucks mostly detected accurately
- Wrong detection of **cars** as trucks (5%)
- Sometimes **bus** goes undetected
- Good performance by SORT, Counting inaccuracy mostly due to detector inefficiency

Vehicle	No. of correct detections	Ground truth	Accuracy (%)
Truck	47	50	94
Car	250	272	92
Bus	19	23	82

⁻ 3 Tasks of Mobility Analysis

HEATMAP

VEHICLE COUNTING

SPEED ESTIMATION

- Approach
 - 1. Detection- YOLOv3
 - 2. Tracking- SORT

- Approach
 - 1. Detection- YOLOv3
 - 2. Tracking- SORT
 - 3. Speed Calculation

3.1 Speed = (**Distance in image)** / (Elapsed time)



- Approach
 - 1. Detection- YOLOv3
 - 2. Tracking- SORT
 - 3. Speed Calculation

3.1 Speed = (**Distance in image)** / (Elapsed time)





- Approach
 - 1. Detection- YOLOv3
 - 2. Tracking- SORT
 - 3. Speed Calculation

3.1 Cross-ratio computation



- Approach
 - 1. Detection- YOLOv3
 - 2. Tracking- SORT
 - 3. Speed Calculation

3.1 Cross-ratio computation in image and world





- Approach
 - 1. Detection- YOLOv3
 - 2. Tracking- SORT
 - 3. Speed Calculation
 - 3.1 Cross-ratio computation in image and world
 - 3.2 Speed = (Distance in real world) / (Elapsed time)



Speed Estimation - Critical Remarks

- Validation unfinished => unavailability of the **ground truth**
- Violation of **collinearity** => some inaccuracy in the results
- More reliable and robust => **Camera Calibration** approach

ANALYSIS AND WEB APPLICATION



Parking Space Usage by Day

- Parking spaces are Mondays and saturday have a larger daily average of 9
- On Saturday, parking spaces are free with a lot of change

Recommendation:

There is no parking space problem overall



Parking Space Usage by Times of Day

- Monday to Thursday, there is an increased parking space usage, starting from 6:59 gradually increasing until 8:20.
- On Tuesday, additionally, an increase between 11:30 and 12:30 can be seen.
- In no point of time, all the spaces were occupied.



Measured Speed by Days

- Saturday 17% higher average speed measured
- Consistency on busses' speeds throughout the measured days.
- Saturday the road is used to drive through instead of being a stopping location.



Measured Speed by Times of Day

Observation:

• From 06:46 to 08:20, drivers tend to speed more on Monday to Thursday

Recommendation

• More speed checks in weekday mornings



Heatmap for Pedestrians

- Made with Matplotlib library of Python
- Pedestrians tend to walk near the wall on the left sidewalk
- No pattern of pedestrians on the street.

Recommendations:

- A crosswalk can be built
- Larger sidewalks on the left side



Heatmap for Bicyclists

- Bicyclists often drive on the left pedestrian sidewalk
- Bicyclist use the bicycle parking station on the right

Recommendations:

• Create a bicycle lane



50

20

Application

- Python **Dash** library
- Wrapped in a Docker
 container for deployment
- Two Screens: Parking spaces and Object tracking results



https://youtu.be/dblebzTdGIY

Application

- Interactive heatmap for pedestrians and bicyclists
- Front-end filtering with
 Plotly plots
- Daily and weekly aggregations for speed and parking spaces measurements



https://youtu.be/SiO5ko1QPmY

Conclusion

- Close collaboration with customer lead to focussed and fast development
- Optimized Object Detection through Retraining
- High demand on computational power and time
- Statistically reliable recommendations possible with more data throughout a year

Future steps:

- Integration of all algorithms possible
- Extend functionalities to
 - narrow zone detection betw. bicyclists and vehicles OR COVID-19 1.5m distance
 - citizen app for parking occupancy prediction
 - Parking lot analysis more accurate if camera is placed differently and parking lots are marked



THANK YOU ! Questions?



Backup Slides

YOLOv3 Benchmark

	backbone	AP	AP ₅₀	AP ₇₅	APS	AP_M	AP_L
Two-stage methods							
Faster R-CNN+++ [5]	ResNet-101-C4	34.9	55.7	37.4	15.6	38.7	50.9
Faster R-CNN w FPN [8]	ResNet-101-FPN	36.2	59.1	39.0	18.2	39.0	48.2
Faster R-CNN by G-RMI [6]	Inception-ResNet-v2 [21]	34.7	55.5	36.7	13.5	38.1	52.0
Faster R-CNN w TDM [20]	Inception-ResNet-v2-TDM	36.8	57.7	39.2	16.2	39.8	52.1
One-stage methods							
YOLOv2 [15]	DarkNet-19 [15]	21.6	44.0	19.2	5.0	22.4	35.5
SSD513 [11, 3]	ResNet-101-SSD	31.2	50.4	33.3	10.2	34.5	49.8
DSSD513 [3]	ResNet-101-DSSD	33.2	53.3	35.2	13.0	35.4	51.1
RetinaNet [9]	ResNet-101-FPN	39.1	59.1	42.3	21.8	42.7	50.2
RetinaNet [9]	ResNeXt-101-FPN	40.8	61.1	44.1	24.1	44.2	51.2
YOLOv3 608 × 608	Darknet-53	33.0	57.9	34.4	18.3	35.4	41.9



Our Vision of Mobility Analysis



Vehicle Counting Analysis



Speed and Parking Space Aggregations

	Weekday	Class	Speed
0	Monday	bus	31.696429
1	Monday	car	29.562825
2	Monday	truck	29.477477
3	Tuesday	bus	31.071429
4	Tuesday	car	29.050574
5	Tuesday	truck	30.756098
6	Wednesday	bus	33.829787
7	Wednesday	car	32.014991
8	Wednesday	truck	32.978261
9	Thursday	bus	31.892857
10	Thursday	car	30.557563
11	Thursday	truck	32.426230
12	Friday	bus	NaN
13	Friday	car	NaN
14	Friday	truck	NaN
15	Saturday	bus	32.771429
16	Saturday	car	36.509188
17	Saturday	truck	37.619048
18	Sunday	bus	NaN
19	Sunday	car	NaN
20	Sunday	truck	NaN

	Weekday	free_parking_spaces	occupied_spaces
0	Monday	9.0	9.0
1	Tuesday	7.0	11.0
2	Wednesday	8.0	10.0
3	Thursday	7.0	11.0
4	Friday	NaN	NaN
5	Saturday	9.0	9.0
6	Sunday	NaN	NaN