

Chapter 10

Energy-Efficient Internet of Things Solution for Traffic Monitoring



Thomas Hoch and Theodorich Kopetzky

Abstract Recent progress in video-based vehicle sensors allows for a detailed observation of road users on intersections in urban areas. By combining the measured real-life traffic situation with thorough traffic simulations, a cooperative system design for the dynamic management of traffic flow including vehicle platoons is possible. In this chapter, we describe our video-based traffic flow estimation system that we installed at a three-way intersection in the small city of Hallein, Austria. We show that the installed system is able to collect comprehensive information about the traffic situation in near real time, and that this information can be used to estimate traffic density and flows of cars and trucks with high precision.

Keywords Edge AI · IoT · Traffic flow analysis · Object detection

10.1 Introduction

The special focus of the Connecting Austria project on the road infrastructure and innovative C-ITS services for level 1 truck platooning as outlined in [20] made the development of a video-based traffic flow estimation system for the assessment of traffic efficiency and safety of platooning in urban areas necessary. The scenario-based evaluation of *use case 4: truck platoon crossing an intersection* is carried out comprehensively for realistic traffic situations at the three-way intersection on the Perner Island in the city of Hallein, Austria (see Fig. 10.1). The investigated intersection was selected because of its mixed traffic that leads frequently to conflicting situations between vehicles, trucks and vulnerable road users like pedestrians or bicycle riders.

A dynamic management of traffic flow [16] including platoons needs not only a precise recognition of the current traffic situation but also a representative statistic over the real traffic situation over the long term in order to optimise for efficiency. This includes the meaningful aggregation of trajectories and the automatic identification of

T. Hoch (✉) · T. Kopetzky
Software Competence Center Hagenberg GmbH, Hagenberg, Austria
e-mail: thomas.hoch@scch.at

© The Author(s) 2022
A. Schirrer et al. (eds.), *Energy-Efficient and Semi-automated Truck Platooning*,
Lecture Notes in Intelligent Transportation and Infrastructure,
https://doi.org/10.1007/978-3-030-88682-0_10

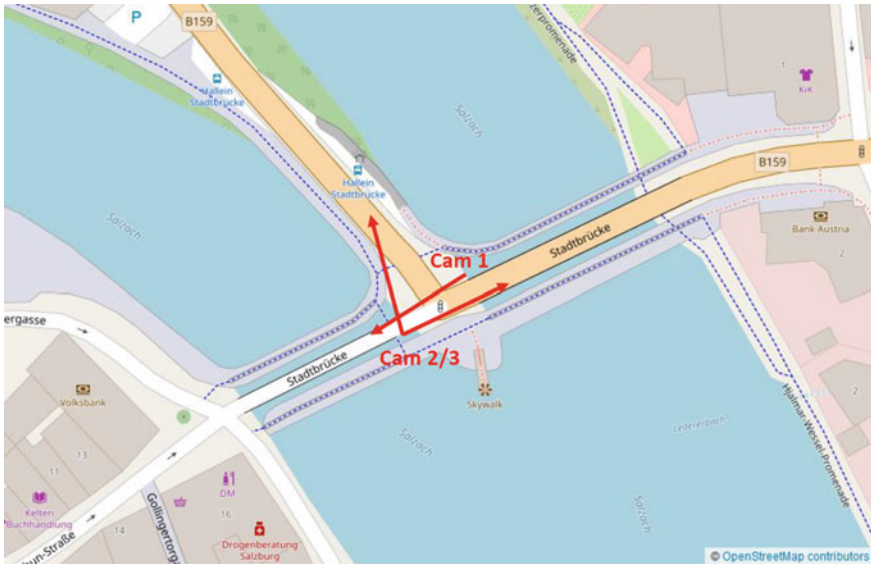


Fig. 10.1 Three-way intersection in Hallein where the traffic monitoring system was installed. The red arrows show the direction of view for each camera, base map and map data from OpenStreetMap, © OpenStreetMap contributors under the CC-BY-SA license, <https://www.openstreetmap.org/copyright>

the flow patterns at the intersection. We therefore developed in cooperation with the project partner *SWARCO FUTURIT Verkehrssignalsysteme GmbH* a video-based traffic measurement system that is able to locate all the traffic participants on the intersection and that can aggregate that information to reveal traffic flow patterns of different road users in great detail. We installed our system at the three-way intersection on the Perner Island in Hallein (see again Fig. 10.1) and tracked six different classes of road users for two weeks in order to get a clear understanding about the real traffic situations and flow patterns on the intersection.

Cameras are widely used for traffic monitoring at intersections and provide rich visual information about road users. A typical set-up consists of a single camera high mounted at a close by building such that the full crossing is observable [6, 24]. This set-up has the advantage that occlusions of road users through other road users are minimised due to the elevated view point and that the configuration process is simplified because the stitching and calibration of different views are not necessary. However, the installation of such a system can become cumbersome since the mounting of the system on a nearby building involves the house owner in the installation process or is not possible at all if the crossing is in a rural area. We therefore follow a different approach where we construct self-contained recognition units that can be easily attached to any traffic light system.

In addition, we were also looking for a more sustainable set-up in terms of energy consumption. There is a growing concern that the strong increase of energy con-

sumption of the IT infrastructure, with the application of deep learning methods as one of the key drivers of this development, at this pace is not sustainable [22]. A currently widely used set-up for street monitoring cameras is to broadcast the video to the cloud and to do the processing of the video stream there. However, the transportation of the high-resolution video data is energy intensive [5] and also needs an Internet connection with high bandwidth in order to not risk processing instabilities. In the Connecting Austria project, we took a different approach taking the energy consumption of the whole system from the beginning of the design process into account. We designed our recognition units as edge computing [26] devices that are able to process the video stream in real time due to a dedicated low energy hardware accelerator for neural networks. Similar solutions have been recently proposed by [2, 17]. Different to their approach, we go one step further and process and analyse the behaviour of the road users in terms of real-world coordinates which allows us to map the extracted traffic patterns to a digital twin of the intersection.

In the following, we describe the developed traffic estimation system that is able to collect comprehensive information about the traffic situation in real time to estimate traffic density and flows of cars and trucks with high precision.

10.2 Low Energy Internet of Things Traffic Monitoring System

Our project aim was the development of a low energy camera-based traffic monitoring system that is able to recognise six different types of road users in real time. The designed system should be easy to deploy on existing traffic light installations and should be able to send the precise location of the road users at the intersection to an operator or to an back-end solutions that can use this information to initiate further actions like warning a driver.

Figure 10.2 shows our measurement set-up at the intersection where we installed one recognition unit for each arm of the three-way intersection. Every recognition units consists of a camera that is connected to a processing unit with a dedicated hardware accelerator for artificial intelligence applications. To preserve the privacy of the road users, the processing of the camera stream is done locally via the attached AI processing unit. Therefore, possible sensitive information never leaves the device, and only the object class and its position in the image are sent to our cloud server. As an information broker, we use a Kafka server,¹ which is an open-source distributed event streaming platform often used in Internet of Things (IoT) scenarios. At our cloud server, the final processing includes three steps: first, the integration of each camera view into one common world view, second, the tracking of the objects over time, and third, the estimation of traffic flow according to our flow graph of the street crossing. Figure 10.3 shows exemplary the intermediate results of our processing pipeline. All processing steps are detailed in the following sections.

¹ <https://kafka.apache.org/>.

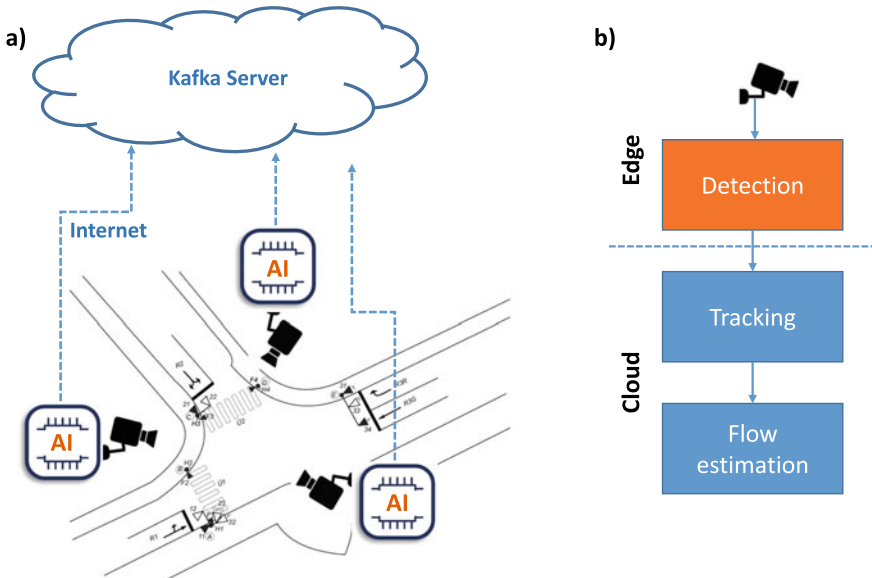


Fig. 10.2 **a** Measurement set-up at the three-way intersection in Hallein. **b** The processing pipeline. Object detection is done with a specialised AI processing unit attached to each of the three cameras. The detection results are transferred via a mobile Internet connection to the cloud infrastructure where the flow estimation is processed

10.2.1 Real-Time Object Detection

The last years showed tremendous progress in the field of object detection. One key driver was the development of new methods and tools that can be efficiently calculated on modern graphic cards. The now standard models like faster RCNN [19] or mask RCNN [8] achieve high accuracy but also need powerful server solutions for the processing of live video streams. Therefore, lightweight models like Yolo [18] or MobileNet [10] have been developed for the use on smartphones or embedded devices. Our system utilises an advanced architecture [25] derived from MobileNet that can be efficiently evaluated on the coral board² that we use to process the live stream of the cameras.

The coral board is an ARM-based single board computer with an on-board Edge TPU co-processor to perform fast machine learning (ML) inferencing. Although the board comes with an object detector, its performance for our setting is rather poor because of the limited images size of 300×300 pixels. In order to increase the detection performance of our system, we increased the image size to 960×384 pixels and trained a new model using a selection of images from the COCO data set [14] and images that we collected at the crossing. The final training data set

² <https://coral.ai/>.

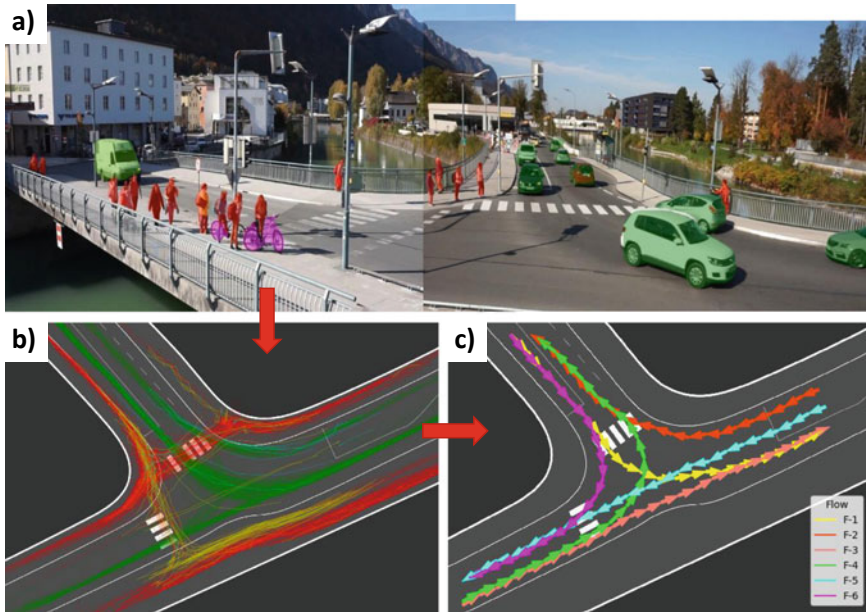


Fig. 10.3 **a** Object detection on camera images with 15 fps where we extract only the bounding box of objects. **b** The detected objects of each camera are transferred to lat/long world coordinates and via Kalman filtering we integrate them to object trajectories. **c** In a final step, the car trajectories are used to automatically infer the major traffic flows on the three-way intersection. In this case, six major traffic flows F-1 to F-6 are identified and depicted in different colours

contained approximately 20k images from the COCO data set and 10k images from the crossing in Hallein that we automatically annotated using the consensus estimate of two state-of-the-art networks [19, 23]. The final model was able to process images of size 960×384 with 15 frames per second (fps).

Several models were trained with the TensorFlow object detection toolkit³ and quantised to 8 bit for the usage on the Coral device. The final model had a mean average precision of 82.6% on a manually labelled holdout data set consisting of 50 images from each of the three cameras at the crossing.

10.2.2 Sensor Fusion and Object Tracking

One important step in the configuration of our system is the projection of the image positions of recognised objects to the coordinate system of the crossing and to combine the projections of each camera to a common world view. Several techniques for roadside camera calibration are available [11] with the overconstrained approaches

³ https://github.com/tensorflow/models/tree/master/research/object_detection.

Table 10.1 Class-specific relative position of the reference point in regard to the detection’s bounding box

Class	Person	Car	Bicycle	Motorecycle	Bus	Truck
f_x	0.28	0.41	0.41	0.44	0.54	0.56
f_y	1.0	0.78	0.89	0.85	0.82	0.83

usually performing best. An accurate camera calibration facilitates the fusion of the projected camera positions and thus the object tracking as a whole. The final sensor fusion and tracking pipeline was implemented the following way.

First we calculated a class-specific reference point P_{ref} from the bounding box of every object detection as shown in the following equation

$$P_{\text{ref}}(x, y) = (x_1 + f_x \cdot (x_2 - x_1), y_1 + f_y \cdot (y_2 - y_1)), \quad (10.1)$$

where x_1, y_1 are the top left coordinates and x_2, y_2 the right bottom coordinates of the bounding box, respectively. The factor f takes a value between $[0 - 1]$ and was optimised in such a way that the distance between the projected world coordinates of the same object seen from different camera views becomes a minimum. The derived values for f_x and f_y are shown in Table 10.1.

Second, with the help of the camera calibration toolbox from the `opencv4` library, we projected the image coordinate of the reference point to our world coordinate system. Each camera was calibrated manually using the visible subset of 20 carefully selected and mapped reference points on the intersection. Third, the projected world coordinates are then assigned to the predicted positions of tracked objects. For every tracked object, there can be maximal one assigned detection per camera. We used the Hungarian algorithm [12] to calculate an assignment with minimal distance between detections and tracked objects. For the Kalman filter update, we used the average position of the assigned detections. The Kalman filter was initialised with the discrete constant white noise kinetic model [4] that we parameterised for every object class individually.

10.2.3 Traffic Flow Estimation

In order to get a better understanding of the vehicle flows on the observed intersection, we used trajectory clustering [3] to group similar trajectories together and to automatically learn the possible traffic patterns at the intersection using an graph-based approach for traffic flow extraction [9]. The method first uses all trajectories to build a flow graph and then extracts flow patterns based on the maximum flow between to nodes of the graph. The major traffic flows of vehicles on the three-way

⁴ <https://opencv.org/>.

Table 10.2 Evaluation result of the traffic flow estimation

Class	Hour	True count	Est. count	Flow combined		Flow pattern	
				Deviation	Error (%)	Deviation	Error (%)
Car	12–13	1244	1286	42	3.4	138	11.1
	14–15	1186	1134	52	4.4	124	10.5
Truck	12–13	72	64	8	11.1	20	27.8
	14–15	56	54	2	3.6	14	25.0

The table shows the error in the count estimate averaged over all six traffic flow patterns combined and separated

intersection are depicted in Fig. 10.3c. The measured vehicle trajectories are mapped to one of the six paths based on the minimal average distance between trajectory points and flow path. Because of tracking errors due to object occlusions or alignment failures, a significant number of tracks could be only observed partly and thus had to be removed from the analysis in order to get a good estimate of the vehicle counts. This was done with an additional calibration step that excluded short tracks due to tracking problems. We used two one-hour recordings with ground truth of the car and truck count to adapt the counting method parameters and measure the flow estimation quality in a cross-validation setting.

10.3 Traffic Flow Measurement Result

First we evaluate our traffic flow estimation method with manually obtained car and truck counts at the three-way intersection. Table 10.2 summarises the true and the estimated count of cars and trucks over two one-hour observation periods. For every observation period, we used the other one to perform a hyper-parameter optimisation of our estimation method. The measured deviation from the true count averaged over the two observation periods was 3.9% for cars and 7.4% for trucks, respectively. Second we also evaluated the precision of the object count for the six observed traffic flow patterns at the intersection (see Fig. 10.3c) individually. The average error of the count estimate increased to 10.8% for cars and to 26.4% for trucks. The strong increase of the error for the truck class is a result of the general low frequency of trucks for flows F-3 to F-6 which makes a precise estimation of the traffic flow more unreliable.

Furthermore, we investigated the traffic flow at the intersection for a two weeks observation period. Figure 10.4 shows the estimate flow of cars and trucks over a 12h time span on an hourly basis. The measurement was done from 18 September till 1 October 2020. The estimates are calculated separately for workdays (orange bars) and weekends (blue bars) since the amount of truck traffic changes considerable during weekends. Whereas the truck traffic has its peak during the morning hours and decays considerably in the afternoon and evening hours, the car traffic stays

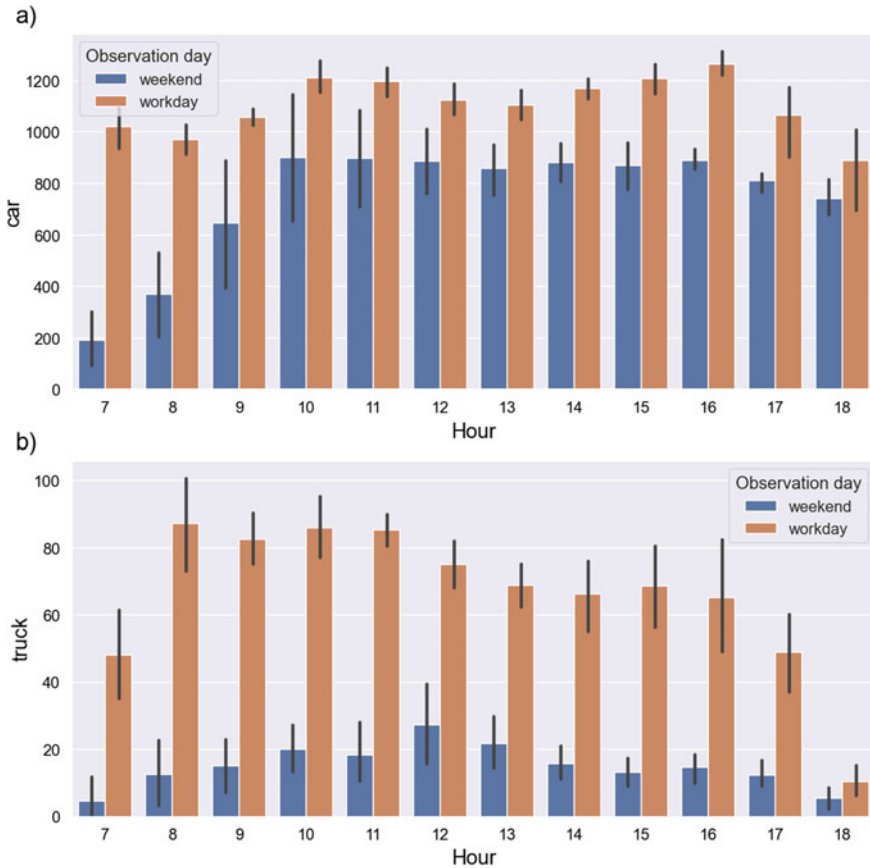


Fig. 10.4 Object count estimates of cars (a) and trucks (b) from 7am to 7pm on an hourly basis. The bars show the mean count of the object class per hour, and the black line indicates the variation of the count estimate. Orange bars are averaged over workdays, whereas the blue bars show the average of weekends

almost constant and decreases only in the evening. The average percentage of trucks on the intersection during this measurement period was 6.0%. During weekends, the average percentage of trucks decreased to 1.8% due to a considerable lower number of trucks passing the intersection (Fig. 10.4b). For cars on the other hand, we find a strong decrease in the count only in the morning hours due to Sundays.

We also investigated how the car and truck traffic is distributed along the six traffic flow patterns. Figure 10.5 shows that most traffic is along flow pattern F-1 and F-2 (see Fig. 10.3c for the definition of the flow patterns) which is a higher ranked street that bypasses the old town. F-3 to F-6 are distributor roads from and into the old town that show less traffic (also because of a smaller time share on the traffic light switching schedule). It can be clearly seen that the truck traffic from and into the

Table 10.3 Percentage of trucks on the intersection during weekdays partitioned by traffic flow patterns

Quantile	F-1	F-2	F-3	F-4	F-5	F-6
0.25	5.4	3.8	1.3	5.0	0.0	0.0
0.75	9.2	7.2	3.7	19.3	3.7	0.0

old town is very low and thus also the percentage of trucks on the intersection as Table 10.3 shows. The only exception is the distributional road from the old town (traffic pattern F-4) for which we observed the highest variation in the percentage of trucks on the intersection. Although the number of cars and trucks along this flow pattern is generally low and leads to high variation in the estimate, Fig. 10.5b also shows that there is significant truck traffic along this flow pattern that explains the strong increase in the percentage of trucks on the intersection.

10.4 Discussion

The evaluation measurement was done on a sunny day with very similar weather conditions between the two observation periods. It is well known that video-based tracking systems are sensitive to weather conditions like fog or snow [6], and we expect an increase in error for such conditions. Simulation of weather conditions via style transfers as in [15] could in principle help to generate a more precise evaluation of the system. However, in our case it was not possible to run an additional deep learning model on the embedded board due to a limited memory and processing capacity.

One difficulty we observed during the execution of this study was the classification of vehicles into separate car and truck classes. During evaluation, we found a gradual transition from car over van to truck class where it was not always easy to draw a clear border between these classes based on vehicle features. Furthermore, our vehicle classification model produced a rather coarse separation of vehicle classes due to the use of pretrained models for the automatically annotation procedure that are aimed for a more general recognition task. Therefore, to get a more standardized vehicle classification as outlined in [7], it would be necessary to extend the training set generator with a specialised classification network as described in MATLAB® [21].

A key feature of our solution was the simple installation procedure on the traffic light itself. The developed device is self-contained with a build in mobile connection to our cloud service and thus needs no wired connection to the Internet which is usually not available at intersections. Although this gives more flexibility in positioning of the device at the traffic light, we also observed that the smaller distance to the road leads to more occlusion of cars and other road users by big vehicles like trucks and buses. Because of these occlusions, a substantial number of vehicles could be tracked only partly leading to more than one track per vehicle. Another factor contributing

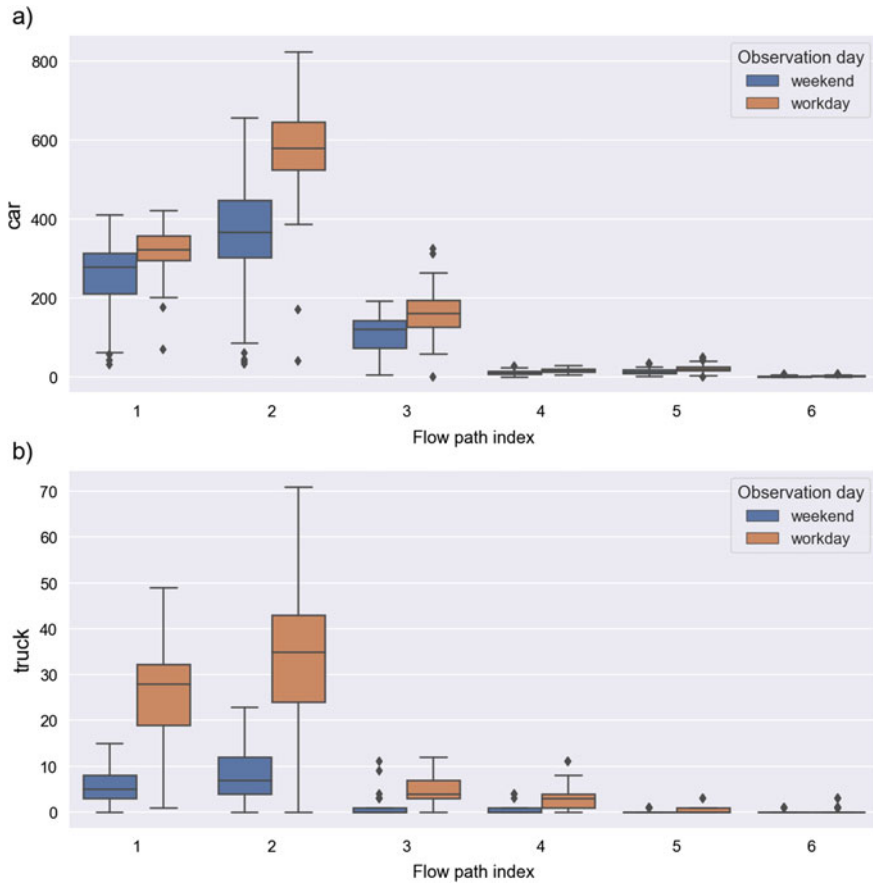


Fig. 10.5 Figure shows the object count statistic of cars (a) and trucks (b) partitioned by flow patterns as defined in Fig. 10.3c. Orange box plots show the summary statistic for workdays, whereas the blue box plots show for weekends, respectively

to this problem is the difficulty of calculating the correct position of vehicles that are only partly seen in the video. In this case, the reference point calculation as described in Sect. 10.2.2 leads to an offset in the position estimate that makes the prolongation of trajectories between camera views more error prone. To circumvent this issue, we had to carefully tune the hyper-parameters of the trajectory selection process. We also observed a higher variation in object size of vehicles that made their recognition more difficult. Since our units were positioned at the roadside, we expect that a more central arrangement above the road could bring advantages.

One specific goal of this investigation was to build a low energy recognition system. Deep learning algorithms are energy hungry [13] and contribute significantly to the increase of energy consumption of the IT infrastructure [22]. Therefore, a sustainable traffic monitoring solution needs to take the energy consumption of the

system into account since a nationwide enrolment would mean the installation of thousands of devices. The presented solution is based on the Coral Edge TPU which provides an energy-efficient way for object detection. The average power used by the device for the video stream processing was approximately 4.9 W (2.4 W in idle mode). Thus with the combination of our technology stack with newly proposed methods for low energy communication in 5G networks [1], an energy-efficient traffic monitoring platform is feasible.

10.5 Conclusion and Outlook

In this chapter, we presented a modern traffic measurement system that has four key advantages over conventional systems: (1) low energy consumption due to edge computing, (2) distributed logic edge and cloud results in a cost-efficient solution, (3) local processing grants a high level of privacy and (4) self-contained field device supports easy on-site installation.

We demonstrated that the system is able to measure the traffic flow of cars and trucks at the three-way intersection in Hallein with high precision and that we are able to partition the vehicle flow into one of the six automatically extracted flow patterns. Our analysis gives more insight on the spatial and temporal distribution of the car and truck traffic at the intersection and provides a basis for a more detailed scenario-based simulation approach in the Connecting Austria project.

In this work, we focused solely on traffic flow measurement of vehicles. The described measurement system can be also used to track and analyse the behaviour of vulnerable road users at an urban intersection. The precise location of these road users could be used to generate C-ITS messages that warn approaching vehicles of potentially dangerous situations as for example “person on the road”. For such a use case, it is critical that the necessary information is provided within a short time frame. Although the measurement frequency of our system with 15 frames per second would in principle allow for such a fast processing, the current design is not favourable for this use case, since the communication with the cloud introduces some significant delays with traditional communication networks. Thus, the latency of such a system is a key factor which we will consider in the future development of our system.

References

1. Al Homssi B, Al-Hourani A, Chavez KG, Chandrasekharan S, Kandeepan S, Energy-efficient IoT for 5G: a framework for adaptive power and rate control. Technical report
2. Barthélemy J, Verstaevl N, Forehead H, Perez P (2019) Edge-computing video analytics for real-time traffic monitoring in a smart city. *Sensors (Switzerland)* 19(9):5
3. Bian J, Tian D, Tang Y, Tao D (2018) A survey on trajectory clustering analysis 2
4. Blackman SS, Popoli R (1999) Design and analysis of modern tracking systems

5. Bolla R, Bruschi R, Davoli F, Cucchietti F (2011) Energy efficiency in the future internet: a survey of existing approaches and trends in energy-aware fixed network infrastructures
6. Datondji SRE, Dupuis Y, Subirats P, Vasseur P (2016) A survey of vision-based traffic monitoring of road intersections 10
7. Hallenbeck M, Selezneva O, Quinley R (2014) Verification, refinement, and applicability of long-term pavement performance vehicle classification rules. Technical report
8. He K, Gkioxari G, Dollár P, Girshick R (2017) Mask R-CNN, 3
9. Hoch T (2021) A spatial knowledge graph for human behavior pattern extraction (in preparation)
10. Howard AG, Zhu M, Chen B, Kalenichenko D, Wang W, Weyand T, Andreetto M, Adam H (2017) MobileNets: efficient convolutional neural networks for mobile vision applications, 4
11. Kanhere NK, Birchfield ST (2010) A taxonomy and analysis of camera calibration methods for traffic monitoring applications. Technical report
12. Kuhn HW (1955) The Hungarian method for the assignment problem. *Naval Res Logistics Q* 2(1–2)
13. Li D, Chen X, Becchi M, Zong Z (2016) Evaluating the energy efficiency of deep convolutional neural networks on CPUs and GPUs. In: *Proceedings—2016 IEEE international conferences on big data and cloud computing, social computing and networking, and sustainable computing and communications*. Institute of Electrical and Electronics Engineers Inc., 10, pages 477–484
14. Lin T-Y, Maire M, Belongie S, Bourdev L, Girshick R, Hays J, Perona P, Ramanan D, Lawrence Zitnick C, Dollár P (2014) Microsoft COCO: common objects in context, 5
15. Luan F, Paris S, Shechtman E, Bala K (2017) Deep photo style transfer 3
16. Nellore K, Hancke GP (2016) A survey on urban traffic management system using wireless sensor networks, 1
17. Nikodem M, Ślabicki M, Surmacz T, Mrówka P, Dołęga C (2020) Multi-camera vehicle tracking using edge computing and low-power communication. *Sensors (Switzerland)* 20(11):1–16
18. Redmon J, Divvala S, Girshick R, Farhadi A (2015) You only look once: unified, real-time object detection
19. Ren S, He K, Girshick R, Sun J (2015) Faster R-CNN: towards real-time object detection with region proposal networks
20. Schildorfer W, Kuhn A, Walter A (2019) Connecting Austria—first results of C-ITS-focused level 1 truck platooning deployment HiTec—an independent non-for-profit research institution into innovation. Technical report
21. Sun W, Zhang X, Shi S, He X (2019) Vehicle classification approach based on the combined texture and shape features with a compressive DL. *IET Intell Transp Syst* 13(7)
22. Vinuesa R, Azizpour H, Leite I, Balaam M, Dignum V, Domisch S, Felländer A, Daniela Langhans S, Tegmark M, Fuso Nerini F (2020) The role of artificial intelligence in achieving the sustainable development goals, 12
23. Wang J, Sun K, Cheng T, Jiang B, Deng C, Zhao Y, Liu D, Mu Y, Tan M, Wang X, Liu W, Xiao B (2020) Deep high-resolution representation learning for visual recognition. *IEEE Trans Pattern Anal Mach Intell*
24. Wang X (2013) Intelligent multi-camera video surveillance: a review. *Pattern Recogn Lett* 34(1):3–19
25. Xiong Y, Liu H, Gupta S, Akin B, Bender G, Kindermans P-J, Tan M, Singh V, Chen B (2020) MobileDets: searching for object detection architectures for mobile accelerators, 4
26. Zhou Z, Chen X, Li E, Zeng L, Luo K, Zhang J (2019) Edge intelligence: paving the last mile of artificial intelligence with edge computing. *Proc IEEE*

Thomas Hoch is a key researcher at the Software Competence Center Hagenberg GmbH for data science. His research interests cover the area of computer vision, scene understanding as well as human behaviour modelling. Thomas Hoch has been engaged in many national and international projects as a principle investigator as for example in the EU projects Tresspass and Teaming.AI.

Theodorich Kopetzky is area manager for Services and Solutions at the Software Competence Center Hagenberg GmbH since the beginning of 2020. Before that he was executive head of the “Knowledge-based Vision Systems” research focus. He has been engaged in many national and international projects, particularly noteworthy as project manager for a multi person-year project in the area of SOA based enterprise application platform systems.

Open Access This chapter is licensed under the terms of the Creative Commons Attribution 4.0 International License (<http://creativecommons.org/licenses/by/4.0/>), which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons license and indicate if changes were made.

The images or other third party material in this chapter are included in the chapter’s Creative Commons license, unless indicated otherwise in a credit line to the material. If material is not included in the chapter’s Creative Commons license and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder.

