

## RULE-BASED MAPPING OF PARKED VEHICLES USING AERIAL IMAGE SEQUENCES

J. Knöttner<sup>2,\*</sup>, D. Rosenbaum<sup>1</sup>, F. Kurz<sup>1</sup>, P. Reinartz<sup>1</sup>, A. Brunn<sup>2</sup>

<sup>1</sup> German Aerospace Center (DLR), Remote Sensing Technology Institute, Wessling, Germany  
- (dominik.rosenbaum, franz.kurz, peter.reinartz)@dlr.de

<sup>2</sup> University of Applied Sciences Würzburg-Schweinfurt (FHWS), Würzburg, Germany  
- julia.knoettner@gmx.de, ansgar.brunn@fhws.de

### ICWG II/III: Pattern Analysis in Remote Sensing

**KEY WORDS:** Aerial Image Sequences, Vehicle Detection, Vehicle Tracking, Parking Space Mapping, Fuzzy Logic

#### ABSTRACT:

Mapping of parking spaces in cities is a prerequisite for future applications in parking space management like community-based parking. Although terrestrial or vehicle based sensors will be the favorite data source for parking space mapping, airborne monitoring can play a role in building up city wide basis maps which include also parking spaces on ancillary and suburban roads. We present a novel framework for automatic city wide classification of vehicles in moving, stopped and parked using aerial image sequences and information from a road database. The time span of observation of a specific vehicle during an image sequence is usually not long enough to decide unambiguously, whether a vehicle stopped e.g. before a traffic light or is parking along the road. Thus, the workflow includes a vehicle detection and tracking method as well as a rule-based fuzzy-logic workflow for the classification of vehicles. The workflow classifies stopped and parked vehicles by including the neighbourhood of each vehicle via a Delaunay-Graph. The presented method reaches correctness values of around 86.3 %, which is demonstrated using three different aerial image sequences. The results depend on several factors like detection quality and road database accuracy.

### 1. INTRODUCTION

Every drive ends sometime, and mostly in a parking lot. Map providers and automobile industry take high efforts to solve the problem of time consuming search for parking spaces. One future example is community-based parking, where vehicles communicate each other by sending free parking space locations. Most effective for detecting parking spaces may be terrestrial or vehicle based sensors, but also airborne monitoring can play a role in providing basis maps for parking spaces even more in the context of the rapid development of drone technology. Moreover, airborne sensors could complete the missing data from terrestrial and vehicle based sensors for example at ancillary roads and suburban regions.

In the last decade, real-time airborne monitoring systems were developed to monitor automatically the traffic flows (Börner et al., 2004, Rosenbaum et al., 2010, Leitloff et al., 2014). These systems extract road traffic information from aerial image sequences and SAR imagery on-board in real-time and to down-link actual traffic information to a ground station via RF-air-to-ground link or a laser link (Horwath, Fuchs, 2009). With this data, emergency authorities and organizations route the individual traffic of the citizens as well as their mobile emergency forces during a major event or disaster.

While automatic traffic data extraction from aerial image sequences already works fine for motorways or in rural areas (Leitloff et al., 2014), urban road traffic data is usually contaminated by cars parking along the roads. Up to now, the problem to separate those parking cars from vehicles, which are participating in traffic but are currently stopped at the moment, when aerial image sequences are taken, was not solved. Other approaches like (Wang, Hanson, 1998) tried to detect parking cars

based on 3D information derived from aerial imagery. Companies like TerraLoup provide maps of parking spaces which are derived from aerial imagery acquired by land survey service. Up-to-now, there is to the best of our knowledge no related work, which exploits the information contained in image sequences to derive parked cars resp. parking spaces. Approaches based on single shot aerial imagery have the problem, that they cannot easily distinguish between moving, stopped and parked vehicles as there is no information about the speed of vehicles.

On the other hand, the time span of observation of a specific vehicle during an image sequence, which takes for example up to 20 seconds corresponding to a usual aircraft flight speed and height, is usually not long enough to decide unambiguously, whether a vehicle stopped e.g. before a traffic light or is parking along the road (Knöttner et al., 2019). In view of this short observation times, we developed a rule-based fuzzy-logic framework to decide about the status of each vehicle. We assume further, that permanent observation of one position for a longer time span for some minutes with e.g. a hovering helicopter would also solve the problem, but this is not useful for bigger areas and not economical at all. Based on the proposed rule-based fuzzy logic framework, the separation between parking and traffic vehicles is then also independent of the flight platform used, e.g. it will be feasible with images from UAVs, aircrafts or helicopters.

In this paper, we propose a new methodology to distinguish between moving, stopping and parking vehicles by exploiting the information contained in aerial image sequences and by using the information of a road database like OpenStreetMap. The distinction step is crucial for a subsequent mapping of parking spaces, which are usually not contained in a road database. First step is the detection and tracking of vehicles, which is described in chapter 2.1. Further, methods to group vehicles in queues (chapter 2.2) and to identify vehicles before crossings

\*Corresponding author

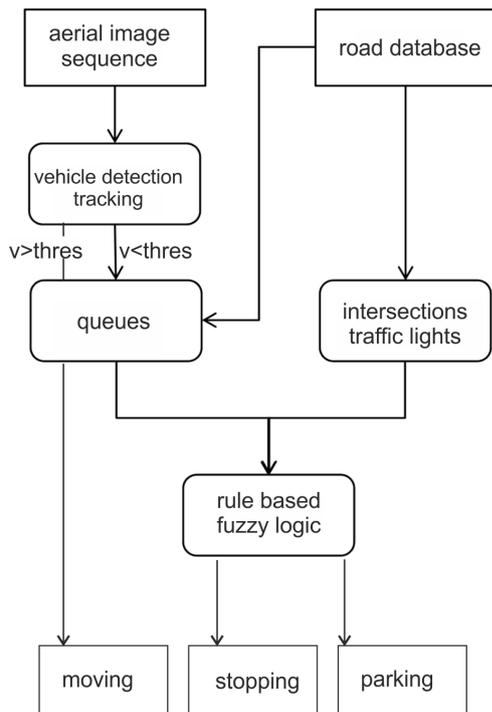


Figure 1. Workflow for vehicle classification

or traffic lights (chapter 2.3) are described. Finally, all relevant parameters are fused together in a fuzzy-logic system to decide, whether a vehicle is participating in the traffic or parking (chapter 2.4). The experiments and the results of this method are shown in chapter 3. Chapter 4 summarizes and concludes the methodology.

## 2. METHODOLOGY

In this section, the overall procedure to distinguish between moving, stopping and parking cars is presented. The first part is the automatic detection and tracking of vehicles, which finally provides the speed and driving direction for each vehicle.

The distinction between moving, stopping or parking vehicles is primarily made under consideration of the neighboring cars. For this reason, all stopping or parking vehicles are grouped into vehicle queues. A vehicle queue is composed of cars that are close to each other on the same road section. Each queue is also parallel to the central axis of the road and all participants have the same direction of travel.

In the next section, a method is described to determine whether the vehicle are part of queues which are participating on the traffic or the queues contain only parked cars, which are standing along a road.

Another valuable information for this distinction are the positions of traffic lights or intersections, which are contained in road databases. Vehicles, which are near to intersections and traffic lights, are distinguished separately.

Finally, all relevant parameters are fused together and a decision is made based on a fuzzy-logic, whether a vehicle is participating on the traffic or a vehicle is parking. Figure 1 shows the general workflow of our approach.



Figure 2. Delaunay triangulation of vehicle positions

### 2.1 Vehicle detection and tracking in image sequences

Automatic robust vehicle detection in remote sensing data is an important machine learning task nowadays and addressed in many articles like (Sommer et al., 2017), (Xia et al., 2018), (Liu, Mattyus, 2015) and (Zhu et al., 2015). The quality can reach more than 90% completeness and correctness depending on the training data used and the properties of the aerial imagery. To derive the moving direction and speed of the vehicles, a tracking based on aerial image sequences must be applied. For this, vehicles detected in the first image of a sequence are tracked in the subsequent images using a shape based matching operator. Since all images should be georeferenced, we can derive directly the position, speed and driving direction of each tracked vehicle. Further details on the applied vehicle detection and tracking are described in (Mattyus et al., 2013) and (Leitloff et al., 2014).

An important parameter is the instantaneous speed of each vehicle  $v$ , which can only be derived from the tracking of vehicles in image sequences

$$v = \sqrt{\Delta Y^2 + \Delta X^2} / \Delta t \quad (1)$$

with  $\Delta X$  and  $\Delta Y$  as differences between coordinates of the vehicle position in two consecutive images and  $\Delta t$  as time difference between two consecutive image acquisition times.

### 2.2 Vehicles in queues

In this section, the identification of vehicle queues is described, which requires information about neighboring vehicles. Therefore, a Delaunay Triangulation is performed, where all vehicle positions are meshed with triangles which have the largest possible interior angles. The result is a Delaunay Graph which consists of edges and vertices (Wessler, Röpcke, 2015). In this case, the vehicle positions are the vertices and the connecting lines between adjacent cars are the edges.

By using the vertices of the Delaunay Graph it is possible to grab the neighborhood for each car. An example is given in figure 2, which shows two vehicles as vertices connected by an edge.

In order to reduce the graph to queues, the road sections, on which the cars are located, must be determined. In a road database, usually each road section is represented as a line with unique Identifier.

The direction of the road segment is also necessary for the final reduction to queues. The angle determines the travel direction on the road section. Vehicles are part of a queue, if all edge



Figure 3. Road axes mapped into a georeferenced image

directions of the graph correspond with the direction of travel on the current road section. Each road section is defined by several points. Road sections without a straight course must be split. Afterwards, the determination of the predetermined direction of travel on each road section is calculated. The following directions play a role in the further analysis:

$$t_{ed,dd,sd} = \arctan(\Delta Y / \Delta X) \quad (2)$$

where  $t_{ed,dd,sd}$  is the direction of the edge  $ed$ , the driving direction  $dd$  or the road segment direction  $sd$  with  $\Delta X, \Delta Y$  as differences between coordinates. The driving direction can be derived from the tracking of vehicles in the image sequences or directly from the orientated bounding box detection of FCNNs as demonstrated in (Azimi et al., 2018). The direction of the road segment can be derived from the road segments stored in the road database (see example in figure 3).

Afterward, the graph can be reduced to vehicle queues. The aim is to keep connections between vehicles which are on the same road section. A neighboring vehicle is then located directly in front of or behind it in the direction of travel. The following criteria are established, which define vehicles belonging to queues:

- $t_{ed} - t_{sd} \leq thres_{angle}$
- $t_{dd} - t_{sd} \leq thres_{angle}$
- Maximum two neighboring vehicles for each vehicle
- Each vehicle should belong only to one queue

The edge direction is the angle from one vehicle to its neighbor. This angle must almost coincide with the given road segment direction. The same applies to the driving direction. We determined an angle threshold of  $5^\circ$  as practicable. For the generation of vehicle queues it is necessary to allow only two neighbors per vehicle and every vehicle should belong only to one queue. All neighbors from a vehicle that do not meet these criteria will be excluded from the existing neighborhood relationship. With the exclusion of single edges vehicle queues are generated which are parallel to the central axis of the road. An example of resulting vehicle queues is illustrated in 4.

### 2.3 Vehicles near to intersections and traffic lights

The further methodology is intended for the classification of vehicles near to intersections or traffic lights. The classification will decide for each vehicle, whether it is participating on the traffic or parking. For this, each vehicle must be referenced to next associated intersection or traffic light. One basic assumption is, that vehicles located nearby intersections or traffic lights are involved in traffic with a higher probability. Both,



Figure 4. Vehicle queues after reducing the Delaunay graph. Queues with moving vehicles are marked green, whereas queues with stopped or parked vehicles are marked red.

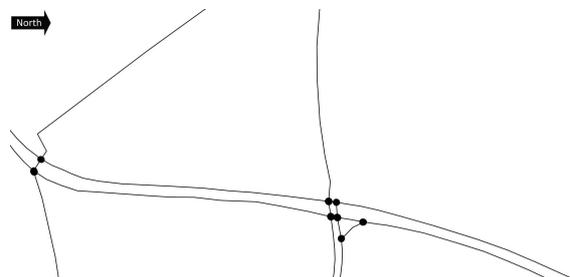


Figure 5. Relevant intersections between roads with higher category.

the positions of traffic lights and intersections can be derived from road databases like OpenStreetMap.

Another assumption is that vehicles on busy roads do temporarily stop more often due to the heavy traffic. For this purpose, the road category is included in the analysis. It is assumed, that the traffic load is directly related to the road category, which is stored in the road database. In figure 5, the relevant road intersection at higher road categories of the example shown in figure 2 are marked with a black dot.

Together with the traffic light positions, it is possible to determine how close vehicles are located to an intersection or traffic light. For this purpose, vehicles are further selected based on the distance to the next intersection or traffic lights. The following distance thresholds are applied:  $36m$  for intersections with two and more lanes and  $17m$  for intersections with one lane in each direction. Figure 6 shows the roads with two and more lanes for each driving direction.

Figures 7 and 8 show all selected vehicles, which are close to in-

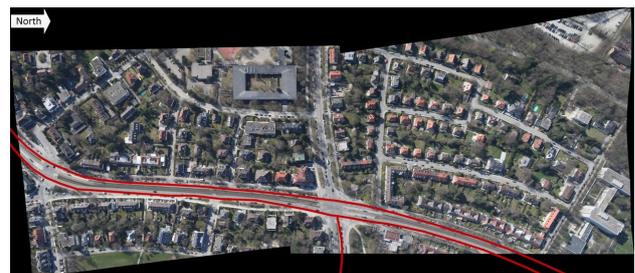


Figure 6. Roads with two or more lanes per driving direction

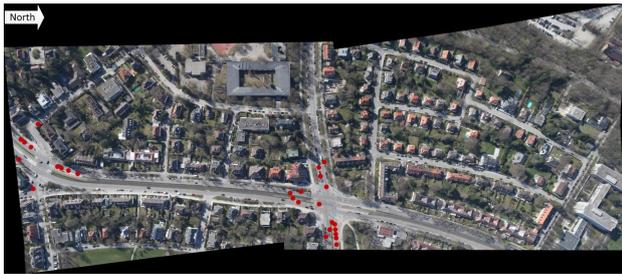


Figure 7. Selected vehicles near to traffic lights.

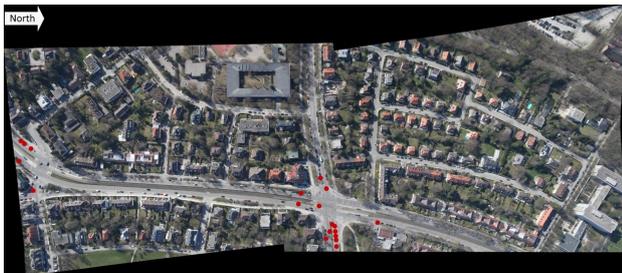


Figure 8. Selected vehicles in areas of crossroads

tersection of traffic lights based on the applied distance threshold.

#### 2.4 Classification of vehicles based on fuzzy logic

The final classification of vehicles in parking and participating in traffic is made by using fuzzy logic, which can be intuitively modelled without complex underlying statistics. In general, vehicles which are assigned to queues and not assigned to queues are processed separately with different fuzzy systems.

Based on predefined rules, several parameters so called linguistic variables are fused together in a fuzzy system. This includes for each vehicle:

- driving speed  $v$
- distance to road intersection/traffic light
- road category
- number of lanes for each driving direction
- vehicle density of a queue  $D$  (only for vehicles in queues)

For example, if a vehicle drives slowly and is located nearby a traffic light or intersection, it is assumed, that this vehicle parks with a lower probability than those vehicles which are not located close to such an area. A fuzzy system receives a sharp input value and returns a sharp output value.

In fuzzification, sharp values are converted into non-boolean values based on not exclusive membership functions. The assignment of a variable to the fuzzy set is done via the membership function. First, the trapezoidal membership functions of the linguistic variables density, speed, distance to road intersection/traffic light and road category are defined as illustrated in figure 9. The definition of the membership functions is done empirically, based on other literature or logical considerations.

The linguistic variable speed has the corresponding terms slow and fast. Even if a vehicle is slowly moving, it cannot automatically be assigned as not parking, as the speed measurement

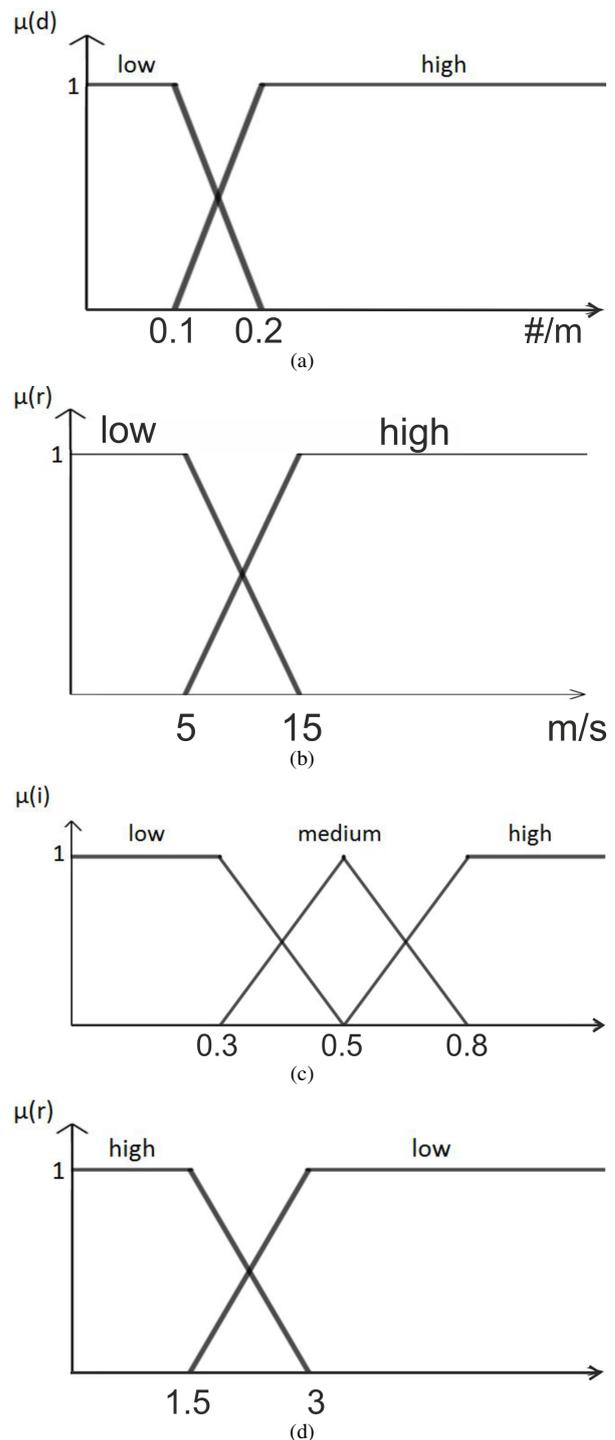


Figure 9. Trapezoidal membership functions of the linguistic variables vehicle density (a), vehicle speed (b), weight parameter of traffic lights or intersections (c) and road category (d).

could be erroneous. The accuracy of the driving speed is influenced by several factors. This includes the accuracy of the vehicle measurement, the timing errors during image acquisition and the scale error caused by the digital elevation model. Taking the individual factors into account, the empirical standard deviation is below 5 km/h (Hinz et al., 2007). Therefore all vehicles that have a driving speed less than 5 km/h can be considered as standing.

Another linguistic variable is the influence by traffic lights or crossroads. Traffic lights and intersections affect the driving behavior of road users. The combinations of them together with the lane category are considered with different weightings in the fuzzy system as listed in 1.

situation	weight
single lane / crossroad	0.5
single lane / traffic light	0.5
single lane / crossroad / traffic light	0.6
twolane / crossroad	0.8
twolane / traffic light	0.8
twolane / traffic light / crossroad	0.9

Table 1. Influence of traffic lights or intersections modelled by a weight parameter.

If the vehicle is close to a traffic light and a crossroad, the impact is higher than with only one existing influence. In addition, the weighting depends on the lane category. If the road consists of two lanes, the road section is more heavily used and the influence of a traffic light or intersection is consequently higher.

The road category is another linguistic variable. The road sections are classified in five functional classes. Common used road categories are listed in table 2.

1	High speed and volume controlled access roads
2	Quick travel between and through cities
3	Moderate speed travel within cities
4	Moderate speed travel between neighborhoods
5	Lower speed travel within neighborhoods

Table 2. Road category (Source: Digital Map Technology, 2017)

The road category is important for the consideration of parking cars along the road in residential areas. Vehicles in those areas receive a special weighting within the fuzzy system. For the definition of the membership function, the road categories 1 – 2 are merged in strong traffic roads and the other categories 3 – 5 in low traffic roads.

Another linguistic variable is the vehicle density of a queue. This variable is only part of the fuzzy system of vehicles in queues. The vehicle density  $D$  is calculated

$$D = n / \Delta s \quad (3)$$

where  $n$  is the number of vehicles within a queue for one edge and  $\Delta s$  is the edge length.

The value provides information about how close vehicles are behind each other. If cars are waiting on a crossroad or traffic

light, the distance between them is small and the density consequently high. It can be assumed that vehicles with a density of 0.15 or above are standing due to traffic (Van Aerde, Rakha, 1995).

After defining the linguistic variables, it is possible to determine the fuzzy values for each vehicle. For linking the facts, if-then rules are defined, which are listed in table 3 for vehicles in queues and in table 4. For each rule, the linguistic variables are linked with “And”.

IF				THEN
Speed	Density	Influence	Road category	Parking
Low	Low	Low	-	high
Low	Low	High	-	Low
Low	High	High	-	Low
Low	High	Low	-	High
Low	Low	Medium	Low	High
Low	Low	Medium	High	High
Low	High	Medium	High	Low
Low	High	Medium	Low	High

Table 3. If-then rules for vehicles in queues

IF			THEN
Speed	Influence	Road category	Parking
Low	Low	-	High
Low	High	-	Low
Low	Medium	High	Low
Low	Medium	Low	High

Table 4. If-then rules for vehicles not in queues.

After the definition of rules the affiliation to the rule base can be determined separately for each vehicle in queues and vehicles not assigned to queues. The dimension of the vector correlates with the number of rules. On this account the degree of affiliation to each rule is the minimum. Then, the accumulation takes place. The vectors “Parking high” and “Parking low” have several fuzzy sets. The accumulation operator combines the partial results with an OR-link to a final fuzzy set. This is realized by T-Conormen. Because of the OR-operation, the maximum operator is used. The output variable is the linguistic variable “parking” with two linguistic terms “low” and “high”. The term low represents “involved in traffic” and the term high “parking”.

In defuzzification, the fuzzy values are converted into sharp values. The output is the parking value with the corresponding validity. All vehicles within a queue receive the same value. All vehicles not in queues receive an individual parking value. Vehicles with a higher parking value than 0.5 are assumed to park, the others are involved in traffic.

### 3. EXPERIMENTS

Experiments were performed using aerial image data obtained from the so called 4k camera system for real-time and rapid mapping applications, a DLR in-house development, which is mounted on a helicopter. The platform used plays no role, as the images acquired from the helicopter have the same characteristics as if they had been acquired from an aircraft.

#### 3.1 Dataset

The method was developed and validated on image sequences of different scenes in the area of Munich, which were acquired with the 4k sensor system at a flight height of 800m above ground. The resulting ground sampling distance was 10cm in nadir direction. The final observation time was less than 10s,

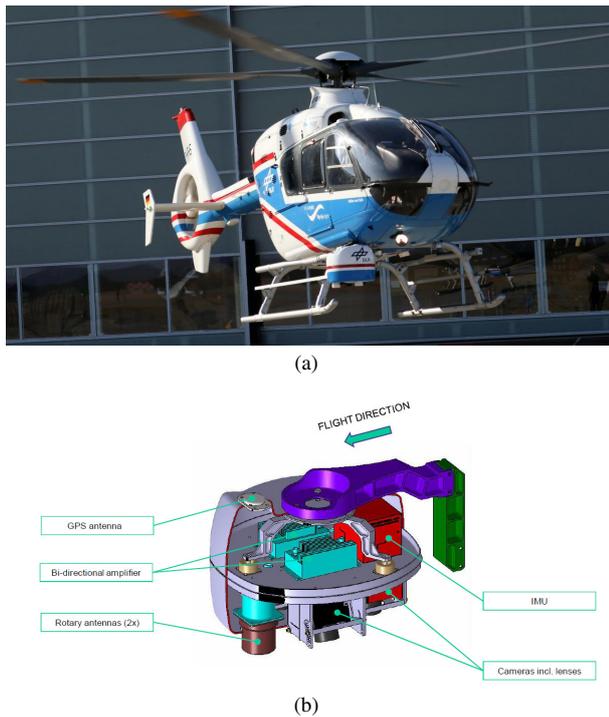


Figure 10. (a) helicopter EC 135 with 4K camera sensor unit. (b) system components of 4K sensor unit.

which is not enough to decide unambiguously, whether a vehicle is parking or not.

The 4k camera system (Kurz et al., 2014) consists of an airborne part and a ground station. We used a BO 105 helicopter as carrier for the airborne system. The airborne system is equipped with a three head optical sensor, a GNSS/IMU navigation unit, industrial PCs for image processing, and a C-band RF-downlink system. The optical sensor (see Fig. 10), mainly consists of three cameras with different looking directions. Cameras are commercial-off-the-shelf types by Canon, namely the Canon EOS 1D-X/C. One camera is aligned to nadir direction, the other ones are the side looking cameras (one to the left, another to the right).

For traffic monitoring, camera time series are usually triggered in burst mode. In this mode, a short image sequence is recorded with a high frame rate. Then, a longer break follows before the next sequence is released. In our case we record three images for each burst with a repetition rate of 2 frames per second and per camera. The duration of the break between each burst is 6-7 sec. These short sequences are fine to detect and track vehicles whereas the breaks of the burst mode reduce the data amount to handle (in real-time) on the PCs of the aircraft system. The flight attitude and GNSS position are measured by an Inertial-Navigation System (INS) which records the data with a frequency of 128 Hz. With this data the software is able to reconstruct the position projected to the ground of each image taken, so that the images can be directly georeferenced.

### 3.2 Results

The images in Figure 11 present the final results of the vehicle classification of one scene. In Figure 11(a), the colors of the vehicle queues correspond to the results of fuzzy logic. All vehicles within a green queue are involved in traffic. A red queue contains parking vehicles. Figure 11(b) presents the final distinction of all vehicles.

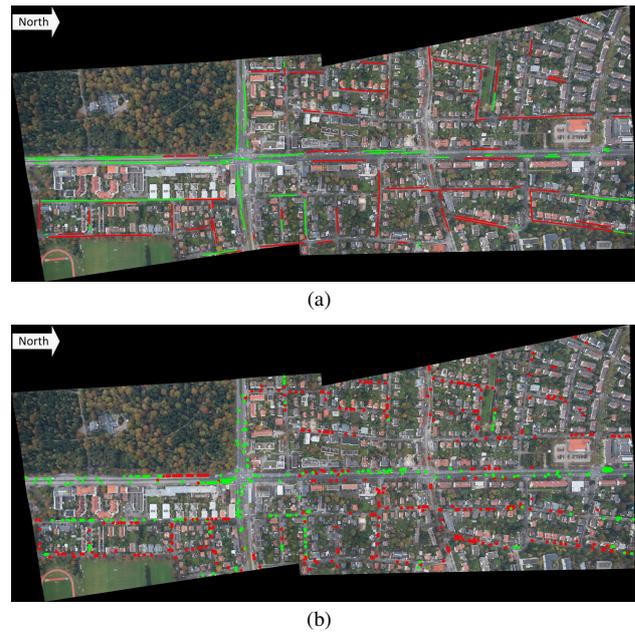


Figure 11. (a) Results for the separation of parking queues and traffic caused stopped queues, (b) Classification results of parking vehicles and traffic caused stopped vehicles

For the evaluation of the parking algorithm, the values for correctness  $Correctness = TP / (TP + FP)$  are calculated for different scenes.

The true positives  $TP$  are the number of parking vehicles detected correctly, false positives  $FP$  are the number of incorrectly determined parking vehicles. In total, images from three scenes were evaluated for the distinction quality. Every scene consists of an image sequence burst, which contains three images acquired at 2 Hz with each camera. Left- and right-looking images were finally stitched together to one image. Ground truth labels were acquired manually for the three scenes.

	True positives	False positives	Correctness (%)
A	769	116	86.9
B	105	16	86.8
C	221	42	84.0
A+B+C	1095	174	86.3

Table 5. Correctness values for three scenes

Table 5 lists the correctness values for the three scenes A, B and C, which reach 86.9 %, 86.8 % and 84 % for the specific scenes. The total correctness is 86.3 %. The parking result depends on chosen vehicle detection and tracking method. The average vehicle detection quality is in the range of 78 % in poor weather to 91 % in good weather conditions. In the scenes investigated, the average quality for tracking was between 88 % and 93 %.

## 4. DISCUSSION AND OUTLOOK

The presented method for classification of vehicles in “moving“, “stopping“ and “parking“ exploits the information contained in aerial image sequences. The time span of observation of a specific vehicle during this sequence is in our datasets not long enough to decide unambiguously relying just on vehicle detection, whether a vehicle stopped before a traffic light or is parking along the road. Besides, the road database gives no further hint, whether the position is a parking place or not.

Thus, we developed a rule-based fuzzy-logic framework to decide about the status of each vehicle.

The presented method reaches a correctness value of around 86.3%. The results depend on several factors, which are discussed in the following.

Clearly, the results depend on the vehicle detection quality. For our experiments, we did not yet use the state-of-the-art deep learning algorithms, but considering the fact that our proposed methodology acts as an extension, the idea and strength of our approach for mapping parked vehicles can be demonstrated as well. The average vehicle detection quality of the applied vehicle detector is in the range of 78% in poor weather to 91% in good weather conditions as validated in (Leitloff et al., 2014).

In the course of neighborhood relationship formations, the final result could be adversely affected from errors in the vehicle tracking. If in a queue, a vehicle has erroneously a movement, all vehicles within this queue will be marked as “traffic participating”.

Furthermore, the result of the performed bundle block adjustment and the georeferencing quality as well is crucial. For the exact determination of the direction of travel and speed, exact image orientations are required.

Another error source is the quality of the road database in terms of completeness of intersections and traffic lights as well as on the quality of road categories and number of lanes. In addition, a high geometrical accuracy of the road database is required. If road axis in the database deviate from the actual central axis of the road, the determined crossroad position will be incorrect. As our approach strongly depends on this information, it would be desirable to become independent of this information in future.

Future developments will be the integration of machine learning algorithms for the whole framework, which includes vehicle detection and tracking within the image sequences. Besides, more appropriate flight configuration, which allow revisiting the same positions are promising to get better results in future.

## ACKNOWLEDGEMENTS

We are grateful to the DLR Flight Experiments facility especially to the crew of the BO 105 helicopter and the DLR Aeronautical Workshop for enabling and realizing numerous flight campaigns during the last four years.

## REFERENCES

Azimi, S.-M., Vig, E., Bahmanyar, R., Körner, M., Reinartz, P., 2018. Towards Multi-Class Object Detection in Unconstrained Remote Sensing Imagery. *CoRR*, abs/1807.02700. <http://arxiv.org/abs/1807.02700>.

Börner, A., Ernst, I., Ruhe, M., Sujew, S., Hetscher, M., 2004. Airborne camera experiments for traffic monitoring. *Proceedings of the XXth ISPRS Congress, 12-23 July 2004, Vol. XXXV, Part B*, 6 p.

Hinz, S., Kurz, F., Wehling, D., Meyer, F., Bamler, R., 2007. Evaluation of Traffic Monitoring based on Spatio-Temporal Co-Registration of SAR Data and Optical Image Sequences. *Photogrammetrie Fernerkundung Geoinformation*, 309–325.

Horwath, J., Fuchs, C., 2009. Aircraft to ground unidirectional laser-comm. terminal for high resolution sensors. *Proceedings of SPIE*, 7199-8.

Knöttner, J., Rosenbaum, D., Brunn, A., 2019. Trennung von parkenden und am Verkehr teilnehmenden Fahrzeugen basierend auf einer automatischen Verkehrserfassung aus Luftbildern. *Publikationen der Deutschen Gesellschaft für Photogrammetrie, Fernerkundung und Geoinformation e.V.*, 28.

Kurz, F., Rosenbaum, D., Meynberg, O., Mattyus, G., Reinartz, P., 2014. Performance of a real-time sensor and processing system on a helicopter. *ISPRS - International Archives*, XL-1, 189–193. <https://www.int-arch-photogramm-remote-sens-spatial-inf-sci.net/XL-1/189/2014/>.

Leitloff, J., Rosenbaum, D., Kurz, F., Meynberg, O., Reinartz, P., 2014. An Operational System for Estimating Road Traffic Information from Aerial Images. *Remote Sensing*, 6, 11315–11341. <https://www.mdpi.com/2072-4292/6/11/11315>.

Liu, K., Mattyus, G., 2015. Fast Multiclass Vehicle Detection on Aerial Images. *Geoscience and Remote Sensing Letters, IEEE*, 12, 1938-1942.

Mattyus, G., Kurz, F., Rosenbaum, D., Meynberg, O., 2013. A real-time optical airborne road traffic monitoring system. *Proceedings Hungarian Association for Image Processing and Pattern Recognition 2013*, 645–656.

Rosenbaum, D., Leitloff, J., Kurz, F., Meynberg, O., Reize, T., 2010. Real-time image processing for road traffic data extraction from aerial images. *Proceedings of the ISPRS Commission VII Symposium 2010*.

Sommer, L. W., Schuchert, T., Beyerer, J., 2017. Fast deep vehicle detection in aerial images. *2017 IEEE Winter Conference on Applications of Computer Vision (WACV)*, 311–319.

Van Aerde, Michel, Rakha, Hesham, 1995. Multivariate calibration of single-regime speed-flow-density relationships. *Proceedings of the Vehicle Navigation and Information Systems Conference (VNIS)*, 334–341.

Wang, X., Hanson, A. R., 1998. Parking lot analysis and visualization from aerial images. *Proceedings Fourth IEEE Workshop on Applications of Computer Vision. WACV'98 (Cat. No.98EX201)*, 36–41.

Wessler, M., Röpcke, H., 2015. *Graphen und Netzwerktheorie - Grundlagen - Methoden - Anwendungen*. Carl Hanser Verlag, München.

Xia, G.-S., Bai, X., Ding, J., Zhu, Z., Belongie, S., Luo, J., Datcu, M., Pelillo, M., Zhang, L., 2018. DOTA: A large-scale dataset for object detection in aerial images. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

Zhu, H., Chen, X., Dai, W., Fu, K., Ye, Q., Jiao, J., 2015. Orientation robust object detection in aerial images using deep convolutional neural network. *Proceedings of the IEEE International Conference on Image Processing (ICIP)*, 2015, 3735–3739.