DETECTION AND TRACKING OF VEHICLES IN LOW FRAMERATE AERIAL IMAGE SEQUENCES*

S. Hinz[1], D. Lenhart[1], J. Leitloff[2]

[1]Remote Sensing Technology, [2]Photogrammetry and Remote Sensing
Technische Universitaet Muenchen, 80290 Muenchen, Germany
{Stefan.Hinz | Dominik.Lenhart | Jens.Leitloff}@bv.tu-muenchen.de

KEY WORDS: Vehicle Detection, Vehicle Tracking, Traffic Monitoring, Traffic Parameters

ABSTRACT:

Traffic monitoring requires mobile and flexible systems that are able to extract densely sampled spatial and temporal traffic data in large areas in near-real time. Video-based systems mounted on aerial platforms meet these requirements, however, at the expense of a limited field of view. To overcome this limitation of video cameras, we are currently developing a system for automatic derivation of traffic flow data which is designed for commercial medium format cameras with a resolution of 25-40 cm and a rather low frame rate of only 1-3 Hz. In addition, the frame rate is not assumed to be constant over time. Novel camera systems as for instance DLR's 3K camera image a scene with "bursts", thereby each burst consisting of several frames. After a time gap of few seconds for readout, the next burst starts etc. This kind of imaging results in an along-track overlap of 90% (and more) during bursts and less than 50% between bursts. These peculiarities need to be considered in the design of an airborne traffic monitoring system. We tested the system with data of several flight campaigns, for which also ground-truth data in form of car tracks is available. The evaluation of the results shows the applicability and the potentials of this approach.

1. INTRODUCTION

1.1 Motivation

Traffic monitoring is a very important task in today's traffic control and flow management. The acquisition of traffic data in almost real-time is essential to swiftly react to current situations. Stationary data collectors such as induction loops and video cameras mounted on bridges or traffic lights are matured methods. However, they only deliver local data and are not able to observe the global traffic situation. Space borne sensors do cover very large areas. Because of their relatively short acquisition time and their long revisit period, such systems contribute to the periodic collection of statistical traffic data to validate and improve certain traffic models. However, often, monitoring on demand is necessary. Especially for major public events, mobile and flexible systems are desired, which are able to gather data about traffic density, average velocity, and traffic flow, in particular, origin-destination flow. Systems based medium or large format cameras mounted on airborne platforms meet the demands of flexibility and mobility. While they have the capability of covering large areas, they can deliver both temporally and spatially densely sampled data. Yet, in contrast to video cameras, approaches relying on these types of cameras have to cope with a much lower frame rate.

An extensive overview on current developments and potentials of airborne and spaceborne traffic monitoring systems is given in (Hinz et al., 2006). In the sequel, we will focus on related approaches that influenced our work to a large extent.

1.2 Related Work

In the last decades, a variety of approaches for automatic tracking and velocity calculation have been developed. Starting with the pioneering work of Nagel and co-workers based on optical flow (Dreschler and Nagel 1982; Haag and Nagel, 1999), the usage of stationary cameras for traffic applications has been thoroughly studied. Further examples for this category of approaches are (Dubuisson-Jolly et al., 1996; Tan et al., 1998, Rajagopalan et al., 1999; Meffert et al., 2005; Kang et al., 2005; Yu et al., 2006). Some of the ideas incorporated in these approaches have influenced our work. Though, a straightforward adoption is hardly possible since these approaches exploit oblique views on vehicles as well as a higher frame rate – both, however, at the expense of a limited field-of-view. Another group of approaches uses images taken by a photogrammetric camera with a high resolution of 5-15cm on ground (e.g., (Zhao and Nevatia, 2003; Hinz, 2004; Punvatavungkour and Shibasaki, 2004)). Also, these approaches are hardly applicable since the vehicle's substructures which are necessary for matching a wire-frame model are no more dominant in images of lower resolution.

In (Ernst et al., 2005), a matured monitoring system for real time traffic data acquisition is presented. Here, a camera system consisting of an infrared and an optical sensor is mounted on slowly moving air vehicles like an airship or a helicopter, but also tests with aircrafts have been conducted. Traffic parameter estimation is based on vehicle tracking in consecutive image frames collected with a frame rate of 5 Hz and more. While the results are promising, a major limitation of this system is the narrow field of view (the width of one single road) due to the low flying altitude that is necessary to obtain a reasonable resolution on ground.

Considering the data characteristics, the most related approaches are (Lachaise, 2005) and (Reinartz et al., 2005, 2006). Like us, they use aerial image sequences taken with a frame rate of 1-3 Hz and having a resolution of 25-40cm. Vehicle detection is done by analyzing difference images of two consecutive frames. This method is quite robust to detect moving objects and to quickly find possible locations for car tracking. Yet, with this approach, it is not possible to detect cars that are not moving, which often also happens for active vehicles if they are stuck in a traffic jam or waiting at a traffic light or stop sign. Furthermore, tracking of detected vehicles includes an interactive component at the current state of implementation.

* Please see CD or online version for color figures of this article.

The boundary conditions of our work are primarily defined by the use of medium format cameras of moderate cost. They allow a large coverage and still yield a resolution of roughly 25cm. However, due to the high amount of data for each image, the frame rate must be kept rather low, i.e. 1 up to a maximum of 3 Hz. Before describing the methodology of detection and tracking in more detail in Sects. 2, 3 and 4, we outline the overall concept of our approach, which is designed to deal with these constraints.

1.3 System Overview

The underlying goal of the concept outlined in the following is the fulfillment of near real time requirements for vehicle tracking and derivation of traffic parameters from image sequences. The general work flow is depicted in Fig. 1.

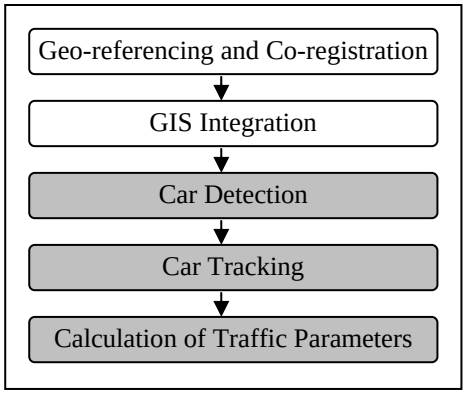

Figure 1. Overall work flow

The images are co-registered and approximately geo-referenced after acquisition. This process is commonly supported by simultaneously recorded navigation data of an INS-/GPS-System. GIS road data, e.g. stemming from NAVTEQ or ATKIS data bases, are mapped onto the geo-referenced images and approximate regions of interest (RoI) are delineated (so-called road sections). Thus, the search area for the following automatic vehicle detection can be significantly reduced.

To acquire data of both flowing traffic and traffic jams, vehicle detection is intentionally separated from tracking; i.e. techniques like optical flow or background estimation by simply subtracting consecutive frames are not included here, since they inherently rely on vehicle velocity as basic feature. Instead – besides of using GIS-road axes as coarse auxiliary data – we incorporate ideas borrowed from automatic road extraction to estimate the local layout and color of a certain road segment under investigation. We focus on color features, since cars on the road may considerably disturb any geometric regularity of the road surface. The final extraction of the cars is done by applying color analysis, dynamic thresholding w.r.t. the estimated road surface, and geometric constraints.

After their detection in the first image, the cars are tracked using image triplets. Since time gaps between frames may get large, tracking is done by matching the cars of the first image (i.e. car patches as reference patches) over the next two images. To this end, an adaptive shape-based matching algorithm is employed including internal evaluation and consistency checks. For each image the reference patch is updated so that illumination and aspect variation are accommodated for. To predict possible locations of previously detected vehicles in the succeeding images a simple motion model is incorporated. This model focuses on smooth tracks and smooth velocity profile, yet with braking cars allowed. From the results of car tracking, various traffic parameters are calculated. These are most importantly vehicle speed, vehicle density per road segment, as well as traffic flow, i.e. the product of traffic density and average speed, eventually yielding the number of cars passing a point in a certain time interval.

Currently, geo-referencing and GIS integration are simulated, thereby accounting for potential impreciseness and uncertainty of the data. The remaining three modules (see gray boxes in Fig. 1) are outlined in the following.

2. VEHICLE DETECTION

The detection process is divided into two stages. In a first step, vehicles with significant color features are extracted by a channel differencing approach. The second step is devoted to detect the remaining gray-scaled vehicles and applies dynamic thresholding constrained to blob-like structures.

Roads normally appear as gray objects in RGB images. In contrast, vehicles may show very strong color information (Fig. 2). In color space, such an object deviates from the gray line while road pixels usually lay close to this line. Therefore, subtracting color channels helps to suppress road structures like lane marks and pronounces colored vehicles (Fig. 2). Initial hypotheses can be extracted by thresholding and unifying the difference images. Simple morphological operations are applied to eliminate clutter and smooth noisy region boundaries. Afterwards, the remaining regions' radii and orientations of the surrounding ellipse are calculated and further false alarms are eliminated using the known road direction and assuming that active traffic must be parallel to this direction. In addition, the radii are used to discriminate between blob-like structures which are supposed to be vehicles and elongated objects which mainly represent road borders or marks. Therefore, the ratio between the semi-major and the semi-minor axis must not exceed an appropriate value. In our tests, the ratio of 4:1 delivered reliable results.

However, there are still many cars that do not show big differences in the spectral channels, so that they are hardly detectable using the scheme above. Therefore, the second part of the extraction focuses on blob-like structures in grayscale channel. Here, instead of substracting the color channels, a dynamic threshold is applied to the grayscale image for detecting initial hypotheses. First, a smoothed version of the original image is calculated by convolution with a Gaussian kernel. Using a minimum threshold (t) for the contrast to the road surface, the condition for regions containing light pixel is $g_o \geq g_t + t$ and for dark pixel $g_o \leq g_t - t$ where g_o represents the original image and g_t the smoothed image. Figure 4 exemplifies the extracted regions for pixels which are brighter than the background. The remaining steps follow as decribed above, i.e. the regions are cleaned and have to fulfill constraints concerning geometry and orientation. Results of the overall detection algorithm are depicted in Fig. 5. It appears that nearly all vehicles are extracted. This result is mainly conditional upon the simple detection and verification process, which even extracts only partial blob-like structures through the use of surrounding ellipses. However, this algorithm also tends to extract a certain number of false hypotheses like shadows. Still, nearly all of these incorrect detections can be eliminated during

the tracking process when comparing their "velocity" to the surrounding objects.

The complete extraction process takes less than 1 second on a normal PC. Reminding of the low frame rate of 1Hz, it can be concluded that the detection results are available even before the tracking starts.

Figure 2. Example of a red car in color image and in the color difference image.

Figure 3. Detection of bright blobs

Figure 4. Example of the automatic car detection

3. VEHICLE TRACKING

In the current implementation of the vehicle tracking, we focus on tracking single cars over image triplets. In addition, GIS-road axes are introduced, which will be referred to as "road polygons" in the sequel. They consist of "polygon points", while two of these enclose a "polygon segment". For each segment, the length as well as the orientation angle are determined.

Figure 5 shows the workflow of our tracking algorithm. Image triplets are used in order to gain a certain redundancy allowing an internal evaluation of the results. Of course, one could use more than three images for tracking. However, vehicles that move towards the flying direction only appear in few images so that the algorithm should also deliver reliable results for a low number of frames. We start with the determination of two vehicle parameters which describe the actual state of a car, namely the distance to the road side polygon and the approximate motion direction (Sect. 3.1). Then, we create a vehicle image model M_C by selecting a rectangle around the car (No. **(1)** in Figure 5). By using a shape-based matching algorithm, we try to find the car in the following images. In order to reduce the search, we select a RoI for the matching procedure based on the motion model (Sect. 3.2). The matching procedure delivers matches M_{12} in Image 1 and the matches M_{13} in Image 3 **(2)**. It should be mentioned, that both M_{12} and M_{13} contain multiple match results also including some wrong matches (see Fig. 5). As output of the matching algorithm, we receive the position of the match center.

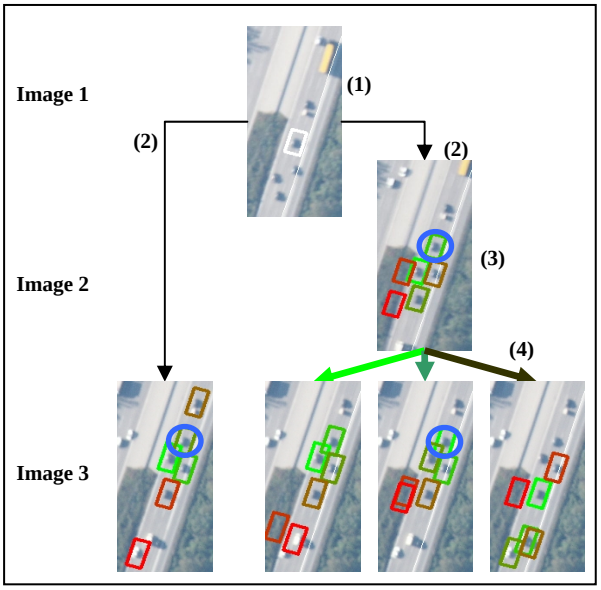

Figure 5. Workflow chart for the vehicle tracking algorithm

For each match M_{12}, vehicle parameters are calculated and new vehicle image models are created based on the match positions of M_{12} **(3)**. These models are searched in Image 3 **(4)**, eventually resulting in matches M_{23}, for which vehicle parameters are determined again. Finally, the results are evaluated and checked for consistency to determine the correct track combination of the matches (marked as blue circle, see Sect. 3.3).

3.1 Vehicle Parameters

The vehicle parameters are defined and determined as follows:

Distance to road polygon: The road polygon closest to a given vehicle is searched, and root point is determined. This point is needed to approximate the direction of the car's motion.

Direction: An initial vehicle's motion direction is approximated as a weighted direction derived from the orientation angles of the three adjacent polygon segments. The weights are inversely proportional to the distance between the car and the end points of the polygon segments. Thus, we also consider curved road segments.

3.2 Matching

For finding possible locations of a car in another image, we are using the shape-based matching algorithm proposed by (Steger, 2001) and (Ulrich, 2003). The core of this algorithm is visualized in Fig. 9. First, a model image has to be created. This is simply done by cutting out a rectangle of the first image around the car's center. The size of the rectangle is selected in such a way that both car and shadow as well as parts of the surrounding background (usually road) is covered by the area of the rectangle.

However, the rectangle is small enough so that no other cars or distracting objects should be within the rectangle. The rectangle is oriented in the approximate motion direction that has been calculated before.

A gradient filter is applied to the model image and the gradient directions of each pixel are determined. For efficiency reasons, only those pixels with salient gradient amplitudes are selected and defined as model edge pixels or model points. Finally, the model image is matched to the gradient image of the search image by comparing the gradient directions. In particular, a similarity measure is calculated representing the average vector product of the gradient directions of the transformed model and the search image. This similarity measure is invariant against noise and illumination changes to a large extent but not against rotations and scale. Hence the search must be extended to a predefined range of rotations and scales, which can be easily derived from the motion model and the navigation data. To fulfill real-time requirements also for multiple matches, the whole matching procedure is done using image pyramids. For more details about the shape-based matching algorithm, see (Ulrich, 2003) and (Steger, 2001).

A match is found whenever the similarity measure is above a certain threshold. As a result, we receive the coordinates of the center, the rotation angle, and the similarity measure of the found match. To avoid multiple match responses close to each other, we limited the maximum overlap of two matches to 20%.

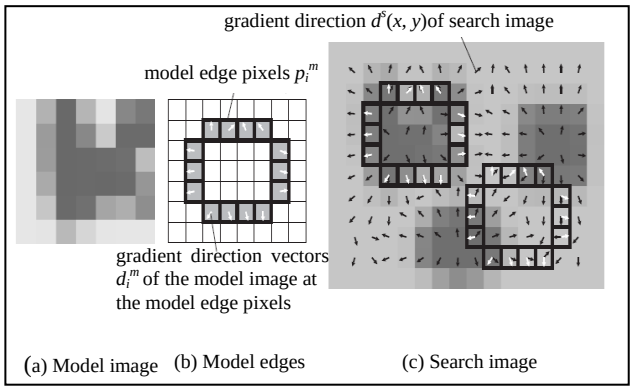

Figure 6. Principle of the shape-based matching method, taken from (Ulrich, 2003), p. 70

3.3 Tracking Evaluation

The matching process delivers a number of match positions for M_{12}, M_{23}, and M_{13}. In our tests, we used a maximum number of the 6 best matches for each run. This means that we may receive up to 6 match positions for M_{12} and 36 match positions for M_{23} for each M_C. Also having 6 match positions for M_{13}, we need to evaluate 216 possible tracking combinations for one car. At a first glance, this seems quite cost intensive. Yet, many incorrect matches can be rejected through simple thresholds and consistency criteria so that the computational load can be controlled easily.

3.3.1 Motion Model: The evaluation incorporates criteria of a motion model for single cars. We suppose that cars generally move in a controlled way, i.e. certain criteria describing speed, motion direction and acceleration should be met. Figure 7 illustrates some cases of a car's movement. For instance, there should be no abrupt change of direction and change of speed, i.e. abnormal acceleration, from one image to the others. In general, the correlation length of motion continuity is modeled depending on the respective speed of a car, i.e., for fast cars, the motion is expected to be straighter and almost parallel to the road axis. Slow cars may move forward between two consecutive images but cannot move perpendicular to the road axis or backwards in the next image.

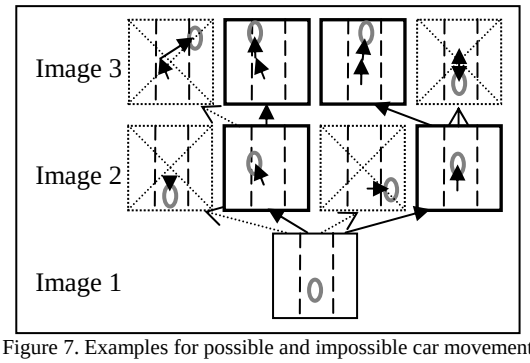

Figure 7. Examples for possible and impossible car movement

3.3.2 Evaluation scheme: As depicted in Fig. 8, we employ a variety of intermediate weights that are finally aggregated to an overall tracking score. Basically, these weights can be separated into three different categories, each derived from different criteria: *i*) First, a weight for the individual matching runs is calculated (weights w_{12}, w_{23}, and w_{13} in Fig. 8). Here, we consider the single car motion model and the similarity measure as output of the matching algorithm which is also referred to as matching score. *ii*) Based on these weights, a combined weight w_{123} for the combination of the matching runs M_{12} and M_{23} is determined. In this case, the motion consistency is the underlying criterion. *iii*) Finally, weights w_{33} are calculated for the combination of the match positions M_{23} and M_{13}. For a correct match combination, it is essential that the positions of M_{13} and M_{23} are identical within a small tolerance buffer.

To avoid crisp thresholds and to allow for the handling of uncertainties, each criterion is mathematically represented as a Gaussian function

$$w(c, \mu, \sigma) = e^{-\frac{(c-\mu)^2}{2\sigma^2}}$$

with the parameters mean μ and standard deviation σ evaluating the quality of an observation with respect to the criterion. By this, the weights are also normalized. In the following, we will outline the calculation and combination of the different weights.

3.3.3 Single Tracking Run: The score w_{match} of the shape-based matching is already normalized (see (Ulrich, 2003) for details). In order to take into account the continuity criterion of a single car's motion, we use the motion directions of M_C and M_{12} to predict an orientation angle for the trajectory from M_C to M_{12}. The difference between the prediction and the actual orientation of the trajectory is used to compute the weight w_{dir}. The combined weight w_{12} then calculates to

$$w_{12} = w_{match} \cdot w_{dir} .$$

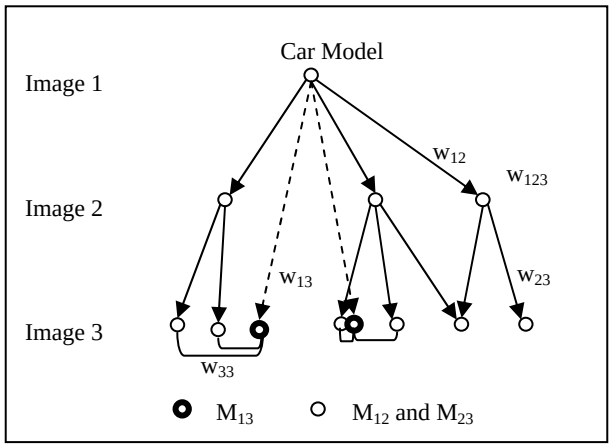

Figure 8. Diagram of the match evaluation process for one car

3.3.4 Motion consistency: In order to exclude implausible combinations of matches, we examine the consistency of a car's trajectory over image triplets. The first criterion of this category is the change of velocity. In typical traffic scenarios accelerations of more than 1.5m/s^2 rarely happen. Again, such values are used to parameterize a Gaussian function resulting in weights w_{vel}. In order to address the continuity of the trajectory, we compare the sum of the distances of the single tracks (M_{12} and M_{23}) with the distance of the direct track to M_{13}. Doing this, we exclude back-and-forth motion of a car. Smaller differences due to bended movement are admitted by the Gaussian weight function which delivers w_{dis}. The weights w_{vel} and w_{dis} are combined to w_{123} by multiplication.

$$w_{123} = w_{vel} \cdot w_{dis}$$

3.3.5 Identity of M_{13} and M_{23}: As a last criterion, the identity of Matches M_{13} and M_{23} is checked (see Fig. 8). Weight w_{33} is simply the distance between the match positions of M_{13} and M_{23} plugged into a Gaussian function.

3.3.6 Final Weight: Assuming that the five individual measurements w_{12}, w_{23}, w_{13}, w_{123}, and w_{33} reflect statistically nearly independent criteria (which, in fact, does not perfectly hold), the final evaluation score W is computed as the product of the five weights:

$$W = w_{12} \cdot w_{23} \cdot w_{13} \cdot w_{33} \cdot w_{123}$$

The correct track is selected as that particular one yielding the best evaluation, however, as long as it passes a lower rejection threshold. Otherwise, it is decided that there is no proper match for a particular car. This may happen when a car is occluded by shadow or another object, but also when it leaves the field-of-view of the images. Fig. 9 shows some results of the tracking over a sequence of 5 images. As it can be seen in the 3rd clip, not every single car could be tracked completely. However, the tracking algorithm delivers a correctness of 100% while the lack in completeness merely results from missing matches.

To track a car through sequences of more than three images, the tracking procedure is continued for further triplets and will be consecutively fed forward by one image. In doing so, we can use the already calculated preceding matches M_{23} and their measures as M_{12}. Based on the car's previous trajectory, we can also roughly estimate the future position to reduce the size of the RoI in the new image, which reduces significantly the computation time for the following tracking. Additionally, this increases the chances that the correct match is found.


Figure 9. Results of tracking over 5 images. Preceding positions are marked by a cross, the current (last) position is marked by the rectangle. The images show only small sections of the entire picture.

4. POST-PROCESSING OF RESULTS

Post-processing the tracking results includes a statistical consistency check for eliminating false alarms and the following computation of traffic parameters.

4.1 Velocity Consistency Check

By analyzing the output of the tracking algorithm, we are able to find further false detections from the car detection procedure. As outlined above, some detections might simply be prominent background structures, which do not move. This knowledge can be exploited depending on the respective scene context. While in city areas, for instance, many cars may move but also many may stand still, such situation is unlikely in other scene contexts. For highway scenes, it can be assumed that either free traffic exists, i.e. *all* cars within a certain neighborhood move with a significant velocity, or congestion may have happened, i.e. all cars are moving very slowly or not at all.

Hence, we calculate the average speed and its standard deviation of all detected cars for a given road section. All cars with a speed outside a 1.5σ-interval are considered as outliers so that a refined mean speed and standard deviation can be derived. Finally, all remaining cars, which have a lower speed than the 3σ border are flagged as inactive and eliminated from further tracking procedure. Please note that slow-moving objects are only eliminated if they are localized at a road segment with dominantly fast moving traffic. In case of dense and slow traffic situations all detections would remain.

A peculiarity of the detection is that often shadows and trailers are detected as separate vehicles. To avoid corruption of the statistical traffic data by these detections, the velocity of cars which are very close together is compared pair-wise. If closed-by pairs move parallel with the same speed, one of them is considered as redundant. In this way also trailers can be merged with the truck since the trajectories are perfectly aligned.

Examples of eliminated cars are marked red in Fig. 11 while merged detections are black.

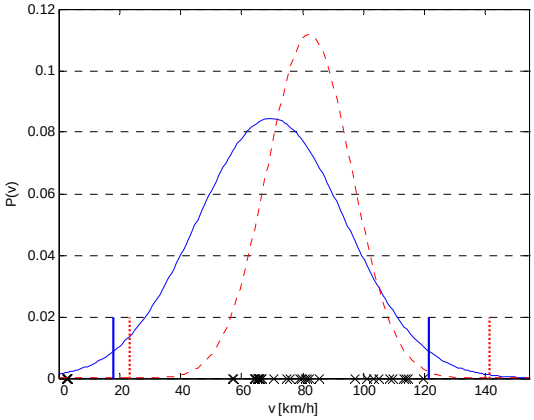

Figure 10. Distribution of the cars' speed. Blue curve and bars: distribution of all cars with 1.5σ–Interval; red dashed curve and bars: distribution of cars within 1.5σ–Interval with 3σ–Interval

Figure 11. Result of false detection elimination. Red boxes mark detections classified as "not moving", black boxes mark detections classified as "redundant".

4.2 Calculation of Traffic Parameters

To calculate traffic parameters, we evaluate the tracked trajectories of the cars. First, we calculate the velocity for each car. This enables us to decide which car is moving on which carriage way. Thereby, we can count the number of cars on each road side. Knowing the length of the road segment which we observed, we can determine the traffic density

$$D = \frac{n}{l}$$

with n being the number of cars on the chosen carriage way and l the length of the road segment. The average speed per carriage way can easily be derived as

$$\bar{v} = \frac{\sum_{i=0}^{n} v_i}{n} \quad \text{with } v_i \text{ being the speed of each car.}$$

In the above experiments, a density of 11 cars per km in each direction and average velocity of 82 km/h (95 and 81 for the other direction) has been determined.

Another important parameter for traffic monitoring is traffic flow. This parameter describes the number of cars passing a fixed position in a certain time interval, which makes it hard to derive directly from aerial images. A vague guess, however, is possible by multiplying the density and the average velocity. While aerial images show advantages to determine the above parameters, induction loops are better suited for calculating the traffic flow. This exemplifies that the analysis of aerial image sequences is a compliant method to already existing measurement methods using stationary sensors.

5. FUTURE WORK

When tracking vehicles in longer image sequences, we are planning to extent the motion model by an adaptive component so that besides evaluating the speed and acceleration of a car, the relations to neighboring cars can also be integrated into the evaluation. This would allow a more strict limitation of the search area and deliver a much more precise measure for tracking evaluation. Another area of research would be the detection and integration of context information such as large shadow areas or partial occlusions to be able to also track vehicles that were partially lost during the tracking.

REFERENCES

Dubuisson-Jolly, M.-P., Lakshmanan, S. and Jain, A. (1996): Vehicle Segmentation and Classification Using Deformable Templates. IEEE Trans. on Pattern Analysis and Machine Intelligence 18 (3): 293–308.

Dreschler, L., Nagel, H-H. (1982): Volumetric model and trajectory of a moving car derived from monocular TV frame sequence of a street scene. CGIP, 20: 199-228.

Ernst, I., Hetscher, M., Thiessenhusen, K., Ruhé, M., Börner, A., and Zuev, S. (2005): New approaches for real time traffic data acquisition with airborne systems. Int. Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences, Vol. XXXVI, Part 3/W24, 69-73.

Haag, M. and Nagel, H.-H. (1999): Combination of Edge Element and Optical Flow Estimates for 3D-Model-Based Vehicle Tracking in Traffic Sequences. Int. Journal of Computer Vision 35 (3): 295–319.

Hinz, S. (2005): Fast and Subpixel Precise Blob Detection and Attribution . Proceedings of ICIP 05, Sept. 11-14 2005, Genua.

Hinz, S. (2004): Detection of vehicles and vehicle queues in high resolution aerial images. Photogrammetrie-Fernerkundung-Geoinformation, 3/04: 201-213.

Hinz, S., Baumgartner, A. (2003): Automatic Extraction of Urban Road Nets from Multi-View Aerial Imagery. ISPRS Journal of Photogrammetry and Remote Sensing 58/1-2: 83–98.

Kang, J., Cohen, I., Medioni, G., Yuan, C. (2005): Detection and Tracking of Moving Objects from a Moving Platform in Presence of Strong Parallax. International Conference on Computer Vision, Vol I, pp. 10-17.

Lachaise, M. (2005): Automatic detection of vehicles and velocities in aerial digital image series. Diploma Thesis, Universitee Lyon.

Meffert B, Blaschek R, Knauer U, Reulke R, Schischmanow A, Winkler F (2005): Monitoring traffic by optical sensors. Proc. of 2nd International Conference on Intelligent Computing and Information Systems (ICICIS 2005): 9-14.

Punvatavungkour, S. and Shibasaki, R. (2004): Three line scanner imagery and on-street packed vehicle detection. International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences, Vol. 35 (B3).

Rajagopalan, A., Burlina, P. and Chellappa, R. (1999): Higher-order statistical learning for vehicle detection in images. Proceedings of the International Conference on Computer Vision, 1999.

Reinartz P., Krauss T., Pötzsch M., Runge H., Zuev S. (2005): Traffic Monitoring with Serial Images from Airborne Cameras. Proc. of Workshop on High-Resolution Earth Imaging for Geospatial Information, Hannover, 2005.

Steger, C. (2001): Similarity measures for occlusion, clutter, and illumination invariant object recognition. In: B. Radig and S. Florczyk (eds.) *Pattern Recognition*, DAGM 2001, LNCS 2191, Springer Verlag, 148–154.

Stilla, U., Michaelsen, E., Soergel, U., Hinz, S., and Ender, J., 2004. Airborne monitoring of vehicle activity in urban areas. International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences, Vol. XXXV, Part B3, 973-979.

Tan, T., Sullivan, G. and Baker, K. (1998): Model-Based Localisation and Recognition of Road Vehicles – International Journal of Computer Vision 27 (1): 5–25.

Toth, C. K., Grejner –Brezinska, D. and Merry, C., 2003. Supporting traffic flow management with high-definition imagery. Proceedings of the Joint ISPRS Workshop on High Resolution Mapping from Space 2003. Hannover, Germany. 6-8 October, (on CDROM).

Ulrich, M., 2003. Hierarchical Real-Time Recognition of Compound Objects in Images. Dissertation, German Geodetic Commission (DGK), Vol. C.

Yu, Q., Cohen, I., Medioni, G., Wu, B. (2006): Boosted Markov Chain Monte Carlo Data Association for Multiple Target Detection and Tracking. International Conference on Pattern Recognition 2006, Vol II, pp. 675-678.

Zhao, T. and Nevatia,R. (2003): Car Detection in Low Resolution Aerial Images. – Image and Vision Computing 21: 693-703.