AUTOMATIC TRAFFIC MONITORING FROM AN AIRBORNE WIDE ANGLE CAMERA SYSTEM

D. Rosenbaum [a, *], B. Charmette [a,b], F. Kurz [a], S. Suri [a], U. Thomas [a], P. Reinartz [a]

[a] German Aerospace Center (DLR), Remote Sensing Technology Institute, PO Box 1116, 82230 Weßling, Germany - (dominik.rosenbaum, franz.kurz, sahil.suri, ulrike.thomas, peter.reinartz)@dlr.de
[b] École Supérieure Chimie Physique Électronique de Lyon, Domaine Scientifique de la Doua, Bât 308 - 43 bd du 11 Novembre 1918, 69616 Villeurbanne Cedex, France - baptiste.charmette@cpe.fr

Commission III, WG III/5

KEY WORDS: Aerial Digital Images, Monitoring, Disaster, Automation, Tracking

ABSTRACT:

We present an automatic traffic monitoring approach using data of an airborne wide angle camera system. This camera, namely the "3K-Camera", was recently developed at the German Aerospace Center (DLR). It has a coverage of 8 km perpendicular to the flight direction at a flight height of 3000 m with a resolution of 45 cm and is capable to take images at a frame rate of up to 3 fps. Based on georeferenced images obtained from this camera system, a near real-time processing chain containing road extraction, vehicle detection, and vehicle tracking was developed and tested. The road extraction algorithms handle a-priori information provided by a road database for a first guess of the location of the roads. Two different techniques can be used for road extraction. In the first method, roadside features are found by using an edge detector based on ISEF filtering, selecting the steepest edge, which is normally the edge between the tarry roads and the vegetation. The second method extracts roads by searching the roadside markings using a dynamical threshold operator and a line detector. Vehicle detection then is limited to the road areas found by the road extraction algorithms. It is based on an edge detector, a k-means clustering of the edges, and on geometrical constraints, e.g. vehicle size. Vehicle tracking is performed by matching detected vehicles in pairs of consecutive images. For this matching the normalized cross correlation is calculated for each detected car within a limited search area. The algorithms for road extraction, vehicle detection and vehicle tracking proved to be quite sophisticated, enabling car detection and tracking rates with a completeness of 70 % and a correctness of up to 90 % on images obtained from a flight height of 1000 m.

1. INTRODUCTION

In a society which relies on plenary mobility, day-to-day large-area traffic monitoring is a quite useful tool to exploit the existing road capacities sufficiently. Moreover, daily commuters are interested in knowing their travel times to work and back, being able to plan their daily business or to change to public transportation systems in case of extensive congestion or traffic jam. Furthermore, during or after mass events and natural disasters, security authorities and organisations as well as rescue forces require fast and sufficient traffic guidance over large areas.

In general, traffic monitoring is mainly based on data from conventional stationary ground measurement systems such as inductive loops, radar sensors or terrestrial cameras. One handicap of these methods is the low spatial resolution depending on their distribution on the ground. New approaches include data by means of mobile measurement units which flow with the traffic (floating car data, FCD, (Schaefer et al. 2002, Busch et al. 2004)). In order to handle traffic monitoring by remote sensing, a number of projects based on optical and SAR airborne sensors, as well as SAR satellite sensors are now running at DLR. One approach currently under development is to use a wide angle optical camera system for near real time traffic monitoring. The big advantage of the remote sensing techniques presented here is that the measurements can be applied nearly everywhere (exception: tunnel segments) and there are no dependencies on any third party infrastructure. Besides, airborne imagery provides a high spatial resolution combined with acceptable temporal resolution depending on the flight repetition rate, but require complex image analysis methods and traffic models to derive the desired traffic parameters. Moreover, estimates for travel times through the area of aerial surveillance can directly be determined from extracted traffic parameters (Kurz et al. 2007b). The paper is arranged as follows. In Chapter 2 the hardware is described as well as the testing data for the processing chain. Chapter 3 characterizes the algorithms used in the processing chain, while Chapter 4 deals with the results from testing this software. Chapter 5 gives conclusions in brief.

2. SENSOR AND DATABASE

2.1 The 3K-Camera System

The 3K-Camera system (3K: "3Kopf" = 3head) consists of three non-metric off-the-shelf cameras (Canon EOS 1Ds Mark II, 16 MPix). The cameras are arranged in a mount with one camera looking in nadir direction and two in oblique sideward direction (Fig 1), which leads to an increased FOV of max *110°/ 31°* in across track/flight direction. The camera system is coupled to a GPS/IMU navigation system, which enables the direct georeferencing of the 3K optical images. Boresight angle calibration of the system is done on-the-fly without ground

* Corresponding author.

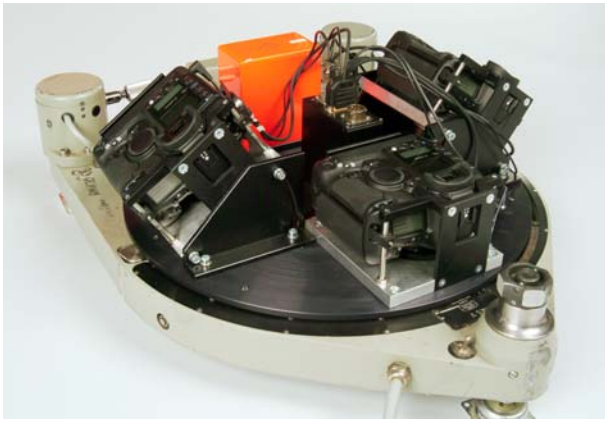

Figure 1. DLR 3K-Camera system consisting of three Canon EOS 1Ds Mark II, integrated in a ZEISS aerial camera mount

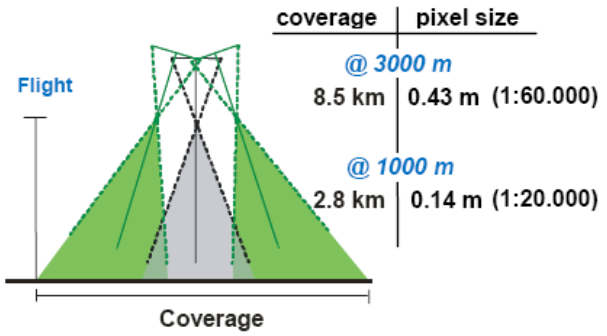

Figure 2. Illustration of the image acquisition geometry. The tilt angle of the sideward looking cameras is approx. 35°

control points based on automatically matched 3-ray tie points in combination with GPS/IMU data (Kurz et al. 2007a).

Fig 2 illustrates the image acquisition geometry of the DLR 3K-camera system. Based on the use of 50 mm Canon lenses, the relation between airplane flight height, ground coverage, and pixel size is shown, e.g. the ground sampling distance (GSD) at a flight height of 1000 m is 15 cm in nadir (20 cm in side-look) and the image array covers up 2.8 km in width.

2.2 Test Site and 3K Imagery

The processing chain was tested on data obtained at the motorways A95 and A96 near Munich, and the "Mittlere Ring" in Munich. The "Mittlere Ring" is a circular main road and serves as the backbone for the city traffic in Munich. It and the adjacent Motorways A95 and A96 are used to full capacity regularly on weekdays during rush hour, and are quite populated all day long. Therefore, these roads are good candidates to find a broad spectrum of traffic situations ranging from free flowing traffic to traffic jam. Hence, they are good targets for aerial images obtained for testing traffic monitoring applications. However, data were taken on 30.04.2007 at noon, which was not during rush hour at all. Data acquisition was performed on two flight strips, one flying ENE, covering the A96 and the western part of the "Mittlere Ring", the other one flying WSW. Thereby, the southern part of the "Mittlere Ring" and the motorway A95 were imaged. The flight height was 1000 m above ground for both strips which leads to a GSD of 15 cm in the nadir camera and up to 20 cm in the side-look cameras. After that, the flight track was repeated at a flight level of 2000 m above ground.
For further analysis, 3K images were geocoded using onboard GPS/IMU measurements with an absolute position error of 3m in nadir images and around one pixel relative. The relative georeferencing error between successive images mainly influences the accuracy of the derived vehicle velocities. Based on simulations and real data, the accuracy of the measured velocity was around 5 km/h depending on the flight height (Hinz et al. 2007).

2.3 Road Database

Data from a road database will be used as *a priori* information for the automatic detection of road area and vehicles. One of these road databases has been produced by the NAVTEQ Company. The roads are given by polygons which consist of piecewise linear "edges," grouped as "lines" if the attributes of connected edges are identical. Up to 204 attributes are assigned to each polygon, including the driving direction on motorways, which is important for automated tracking. Recent validations of position accuracy of NAVTEQ road lines resulted in 5m accuracies for motorways.

3. PROCESSING CHAIN

On the data obtained as described before, the processing chain for traffic monitoring was tested. This experimental processing chain, consisting of several modules can be roughly divided into three major steps. These are road extraction, car detection, and car tracking (see also fig 4).

3.1 Road Extraction

For an effective real time traffic analysis, the road surface needs to be clearly determined. The road extraction starts by forming a buffer zone around the roads surfaces using a road database as described above as a basis for the buffer formation process. In the next step, two different methods for further feature analysis can be applied. Both modules automatically delineate the roadsides by two linear features. One module works as follows: Within the marked buffer zone, edge detection and feature extraction techniques are used. The critical step of edge detection is based on an edge detector proposed by Phillipe Paillau for noisy SAR images (Paillou, 1997). Derived from Deriche filter (Deriche, 1987) and proposed for noisy SAR images, we found this edge detector after ISEF filtering (Shen and Caston, 1992) extremely efficient for our purpose of finding edges along the roadsides and suppressing any other kind of surplus edges and noise present. With this method, mainly the edge between the tarry road area and the vegetation is found. The alternative module searches for the roadside markings by extracting lines on a dynamic threshold image. In this module, only the longest lines are kept representing the drawn through roadside marking lines. As a side effect, the dashed midline markings are detected in this module, too. These markings often cause confusion in the successional car detection, since they resemble white cars. However, these false alarms can be deleted from car detection, since the module for roadside marking detection finds the dashed midline markings and stores them in a separate class.
In a next step, the roadside identification module, again with the help of the road database tries to correct possible errors (gaps and bumps) that might have crept in during the feature

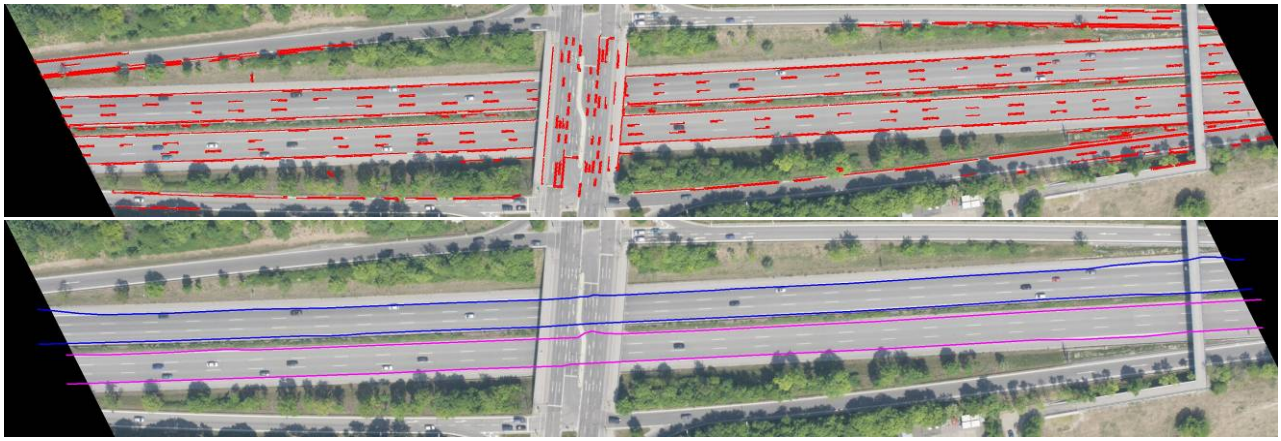

Figure 3. Example for road extraction at A96 exit Munich-Blumenau (clipping from nadir image). Upper panel shows line detections at a flight height of 1000 m, panel below shows the resulting road area after smoothing / gap filling.

extraction phase. Furthermore, it smoothes the sometimes curly road boundary detections from feature extraction (see fig. 3). Gaps due to occlusion of the road surface by crossing bridges are closed, if gapping is not too large. This has the advantage that the course of the road is not lost, although the road itself is not seen at this place. However, it could lead to false alarms in the car detection. If cars are crossing the bridge, they might be assigned belonging to the occluded road below the bridge spuriously in car detection. However, we try to sort them out by alignment, since they are elongated perpendicular to the course on the occluded road.

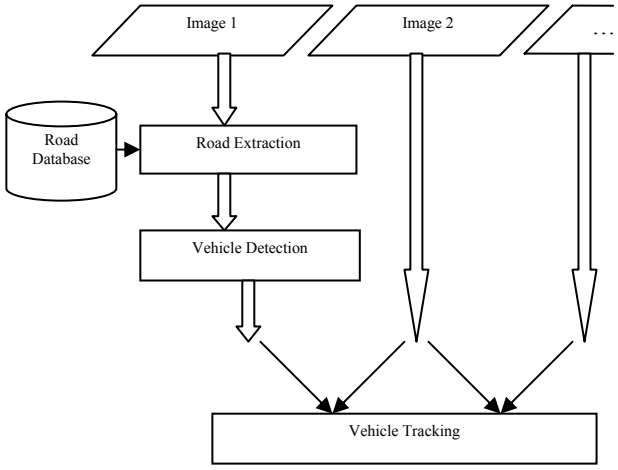

Figure 4. Implemented processing chain for a knowledge based road extraction, vehicle detection, and vehicle tracking on an image sequence.

3.2 Vehicle Detection

With the information of the roadside obtained in the processing step described before, it is possible to restrict vehicle detections and tracking only to the well determined road areas. This increases performance and enhances the accuracy of vehicle detection. Based on this, we developed an algorithm for the detection of vehicles which is described in the following.

With the information about the alignment and direction of the roadside, we are able to mark all pixels belonging to the road. Thereby, the local road direction is included as an extra parameter into the marker of each pixel. For the vehicle detection, a Canny edge algorithm (Canny, 1986) is applied and a histogram on the edge steepness is calculated. Then, a k-means algorithm is used to split edge steepness statistics into three parts which represent three main classes. These three classes are namely edges belonging to vehicles, edges belonging to roads, and edges within road and vehicle edges, and therefore not yet classifiable.

We consider the part with the lowest steepness in the edge histogram being mainly populated by pixels of the road background, since its intensity is quite uniform. Moreover, we assume that the part with the highest steepness -due to the high discontinuity in the intensity- is most likely populated by vehicles. However, this part of the statistic is not only occupied by vehicle edges. It can be also contaminated by midline markings, shadows, sign boards, trees, or the like.

In the histogram part containing the edges not yet classified, it must be determined which pixels belong to the road background or to potential vehicles. For this decision, the pixel neighbourhood is examined. Pixels directly connected with a potential vehicle pixel are moved into the vehicle class. Remaining pixels are finally considered as road background and neglected.

In the next step, the edges belonging to the roadside markings

Potential car pixels detected before closing

Potential car pixels after closing

Figure 5. Closing the shapes of potential car pixels.

still contaminating the vehicle class are eliminated from the histogram. As the roads are well determined by the road extraction, these roadside lines can be found easily. Thus, the algorithm erases all pixels with high edge steepness which are laying on a roadside position. These pixels are considered mainly belonging to the roadside markings. Thereby, the algorithm avoids erasing vehicles on the roadside by observing the width of the shape. Since vehicles are usually broader than roadside lines, this works unproblematic. Midline markings, which were detected by the roadside identification module based on the dynamical threshold image, are erased, too. This is done in order to reduce false detections, since these midline

Figure 6. Examples for vehicle detection on motorways (upper image, A96 exit Munich-Blumenau, clipped nadir exposure) and in the city (lower image, Munich "Mittlerer Ring", clipped side-look-left exposure). Rectangles mark automatic vehicle detections, triangles point into direction of travel.

markings may mock up white cars. Then, potential vehicle pixels are grouped by selecting neighboured pixels. Each region is considered to be composed of potential vehicle pixels connected to each other. With the regions obtained a list of potential vehicles is produced. In order to mainly extract real vehicles from the potential vehicle list, a closing and filling of the regions is performed. This step is shown in fig 5.

Using closed shapes, the properties of vehicle shapes can be described by their direction, area, the length and width. Furthermore, it can be checked if their alignments follow the road direction, and its position on the road can be considered as well. Based on these observable parameters, we created a geometric vehicle model. The vehicles are assumed to have approximately rectangular shapes with a specific length and width oriented in the road direction. Since they are expected to be rectangular, their pixel area should be approximately equal to the product of measured length and width and vehicles must be located on the roads. We set the values for the minimum expected vehicle length to 5.7 m and for the minimum width to 2.6 m. Since, these values are minima constraints on vehicle geometry, we are able to detect cars and trucks. In case of several detections with very low distances the algorithm assumes a detection of two shapes for the same vehicle. Then, it merges the two detections into one vehicle by calculating averages of the positions. Finally, based on this vehicle model, a quality factor for each potential vehicle is found and the best vehicles are chosen.

For traffic monitoring, the camera system is in a recording mode, that we call "burst mode". In this mode, the camera takes a series of four or five exposures with a frame rate of 3 fps, and then it pauses for several seconds. During this pause, the plane moves significantly over ground. Then, with an overlap of about 10 % to 20 % to the first exposure "burst", the second exposure sequence is started. Continuing this periodical shift between exposure sequences and brakes, we are able to perform an area-wide traffic monitoring without producing an overwhelming amount of data. Our strategy for traffic monitoring from this exposures obtained in "burst mode" is to perform a car detection only in the first image of an image sequence and then to track the detected cars over the next images (fig. 4).

3.3 Vehicle Tracking

With a vehicle detection performed on each first image of the image sequences as described above, we are able to track the found cars over the whole image sequence. For that, a template matching algorithm based on a normalized cross correlation operator is performed. For each detected vehicle, a circular template image is cut off the first image of the image sequence at the position of the detected car. Depending on its position and direction of travel (obtained from the road database) in the first image, a search window is created in the second image for each vehicle. Within this search window spanned inside the second image, the created template image is cross correlated to an area of the same size, while it is moved line- and column wise over the search window. The correlation factor is a measure of the probability for a hit. The maximum of the correlation score is stored, if the correlation factor is greater than 0.9. We introduced this border in order to avoid hits, if the car from the first image is not anymore present in the second image. A border of 0.9 turned out to be the best value to track vehicles present in both images, and to avoid errors in tracking if the vehicle is missing in the second image. With the maximum of the correlation score, the image coordinates in the second image as well as the corresponding geocoordinates at the location where the maximum in correlation appeared are stored. This gives the location of the vehicle matched from the first image into the second image. With this position stored for each vehicle, we are able to repeat the application of the car tracking module for an image pair consisting of the second and third image of an image sequence, and afterwards it is applied to the third and fourth image. Hence, we are able to track a detected vehicle over the whole image sequence of one exposure burst. Moreover, we are able to track vehicles from one image sequence into another image burst, if the overlap is sufficient. However, if the overlap is in the range of 10 % to 20 %, only few vehicles are situated in more than one image sequence, so that we consider reasonable to restrict vehicle tracking only to image pairs obtained within one exposure sequence.


Figure 7. Car tracking by normalized cross correlation of a group of three cars detected in the first image of a sequence (right) to the second image (left). Clipped images were taken from the scene shown before at the motorway A 96.

4. RESULTS

We tested our processing chain based on the data take from 30.04.2007 as described in Chapter 2. For that, the completeness and correctness of vehicle detection and tracking are determined on data of several resolutions, obtained from different flight levels.

4.1 Road Detection

Road detection was performed using two different modules. It turned out, that detecting roadside markings for determining the road area is a good strategy on images taken at a lower flight height of 1000 m resulting in a resolution of 15 cm GSD. Nevertheless, at higher flight levels (for instance at 2000 m) road extraction works well with the module searching the edge between blacktop and vegetation. Fig. 3 shows typical results of road extraction using roadside markings. Top image shows the line extraction, whereas in the image below the finally extracted roadsides after smoothing and closing gaps are shown.

4.2 Vehicle Detection

In order to quantify the vehicle detection efficiency, test data were processed and the results of the automatic vehicle detection were compared to a manual car detection. Table 1 shows the results of the comparison between automatic and manual car detection. On a flight height of 1000m (15 cm GSD), vehicle detection performs well on motorways with a correctness of around 80 % and a completeness of 68 %. In a complex scene like the city ring road we can proof that car detection delivers respectable results with a completeness of 65 % and a correctness of 75 %. However, at a flight height of 2000 m (GSD=30 cm) performance drops down to 56 % in completeness but correctness is still high with 76 %.

Site	correct	false	missed	correctness	completeness
Motorway (1000m)	85	22	41	79%	68%
Motorway (2000m)	95	30	76	76%	56%
City (1000m)	47	16	25	75%	65%

Table 1. Results on testing vehicle detection on data obtained at several test sites (from different flight heights). Counts of correct vehicle detections, false alarms and missed detections, as well as correctness and completeness in percentage are given.

Figure 6 shows examples of vehicle detection performed on images obtained at a flight height of 1000 m. Upper image was taken on highway A96 near exit Munich-Blumenau, lower image shows part of the circular road "Mittlerer Ring" in Munich city. Only few false alarms were detected.

4.3 Vehicle Tracking

Vehicle tracking was tested on the same data takes obtained at a flight height of 1000 m (15 cm GSD) and at a flight height of 2000 m (30 cm GSD). Figure 7 shows a typical result on tracking vehicles from the first image of an image sequence into the second exposure of the sequence. On images with a resolution of 15 cm GSD, vehicle tracking on motorways performs perfectly well, with a correctness of better than 95 % and a completeness of almost 100 % on each image pair. On images obtained from higher flight levels (≥30 cm GSD) tracking still works fine with a completeness of 90 % while having a correctness of 75 %. We attribute the good tracking performance on low flight heights to the fact that with a resolution of 15 cm GSD vehicle details like sunroof, windscreen and backlite, and body type go into the correlation which simplifies finding the correct match. However, these details are not anymore seen at higher flight levels.

4.4 Performance

As we are planning to execute the extraction of traffic parameters on the 3 x 16 Mpix rgb-images obtained from the 3K camera in near real time, we have to focus upon performance. Therefore, tests on road and vehicle detection as well as vehicle tracking were performed on actual standard hardware consisting of a dual-core PC with a CPU frequency of 1.86 GHz and 2 GB RAM. Road extraction on a typical motorway takes less than 30 s for one nadir and two side-look exposures in total. Vehicle detection on these three images needs 25 s of calculation time on the present system. In comparison, car tracking is quite fast, consuming only 15 s for a tracking of 3 x 15 cars over an image sequence consisting of 3 x 4 images. Moreover, the pure calculation time for cross correlation is 30 ms per vehicle for a tracking through the whole sequence. In total, it costs 70 s to analyse the traffic within one image sequence. That means, assuming a break of 7 s between each image "burst" (which would result in a overlap of 10 % between two image "bursts" at a flight speed of 60 m/s and a flight height of 1000 m), we have a time overhead in the processing chain for traffic monitoring of a factor of 10. However, the prototype of our processing chain is built up still modular, which means that each module in the chain reads an image from hard disk into memory, performs an operation, and at the end writes a new image to hard disk. We estimate to

halve the overhead by reducing hard disk read/write. Further progress in reducing calculation time could be made by using actual hardware with high performance quad core CPUs.

5. CONCLUSIONS

Despite the large amount of incoming data from the wide angle camera system, we are able to perform traffic data extraction with a high actuality. Thereby, high accuracies for velocities (5 km/h), good correctness in vehicle detection (79 %) and in vehicle tracking (90 % of detected vehicles) is reached. Hence, the investigations show the high potential using aerial wide angle image series for traffic monitoring and similar applications, like the estimation of travel times or the derivation of other relevant traffic parameters. Moreover, the data processing speed can be improved in future by converting the modules of the processing chain into tasks which hand over pointers to images stored in the RAM instead of reading and writing them on hard disk, as done by our prototype.

REFERENCES

Busch, F.; Glas, F.; Bermann, E. (2004). Dispositionssysteme als FCD-Quellen für eine verbesserte Verkehrs-lagerekonstruktion in Städten - eine Überblick. *Straßen-verkehrstechnik* 09/04

Canny, J. F. (1986). A computational approach to edge detection. *IEEE Trans. Pattern Analysis and Machine Intelligence*, Vol. 8 (6), pp 679-698.

Deriche, R. (1987). Using Canny's criteria to derive an optimal edge detector recursively implemented. *The International Journal on Computer Vision,* Vol. 1, No. 2, pp 167-187

Hinz, S., Kurz, F., Weihing, D., Suchandt, S., Meyer, F., Bamler, R. (2007). Evaluation of Traffic Monitoring based on Spatio-Temporal Co-Registration of SAR Data and Optical Image Sequences. *PFG – Photogrammetry – Fernerkundung – Geoinformation,* 5/2007, pp 309-325.

Kurz, F., Müller, R., Stephani, M., Reinartz, P., Schroeder, M. (2007 a). Calibration of a wide-angle digital camera system for near real time scenarios. In: *Heipke, C.; Jacobsen, K.; Gerke, M. [Eds.]: ISPRS Hannover Workshop 2007, High Resolution Earth Imaging for Geospatial Information, Hannover,* 2007-05-29 - 2007-06-01, ISSN 1682-1777

Kurz, F., Charmette, B., Suri, S., Rosenbaum, D., Spangler, M., Leonhardt, A., Bachleitner, M., Stätter, R., Reinartz, P. (2007 b) Automatic traffic monitoring with an airborne wide-angle +digital camera system for estimation of travel times. In: *Stilla, U., Mayer, H., Rottensteiner, F., Heipke, C., Hinz, S. [Eds.]: The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences,* Vol. 36 (3/W49B), pp 83 -86.

Paillau, P. (1997). Detecting Step Edges in Noisy SAR Images: A New Linear Operator. *IEEE Transactions on Geoscience and Remote Sensing,* Vol. 35, No.1, pp 191-196

Schaefer, R.-P., Thiessenhusen, K.-U., Wagner, P. (2002). A traffic information system by means of real-time floating-car data. *Proceedings of ITS World Congress,* October 2002, Chicago, USA.

Shen, J., and Castan, S. (1992). An optimal linear operator for step edge detection. *CVGIP, Graphics Models and Image Processing,* Vol. 54, No. 2, pp 112-133