An extended version of this paper has been published. Please cite instead:

Knoop, V.L., Hoogendoorn, S.P. and Van Zuylen, H.J. (2009) Processing Traffic Data collected by Remote Sensing. Transportation Research Records: Journal of the Transportation Research Board, No. 2129, pp. 55-61

Processing Traffic Data collected by Remote Sensing

Victor L. Knoop MSc
(corresponding author)
TRAIL research school PhD Student
Delft University of Technology
Faculty of Civil Engineering
Transport & Planning
P.O. Box 5048
2600 GA DELFT
The Netherlands
Tel. +31 15 27 81723
Fax +31 15 27 83179
E-mail v.l.knoop@tudelft.nl

Serge P. Hoogendoorn PhD
Professor
Fellow TRAIL research school
Delft University of Technology
Faculty of Civil Engineering
Transport & Planning
P.O. Box 5048
2600 GA DELFT
The Netherlands
Tel. +31 15 27 85475
Fax +31 15 27 83719
E-mail s.p.hoogendoorn@tudelft.nl

Henk J. van Zuylen PhD
Professor
Fellow TRAIL research school
Delft University of Technology
Faculty of Civil Engineering
Transport & Planning
P.O. Box 5048
2600 GA DELFT
The Netherlands
Tel. +31 15 27 82761
Fax +31 15 27 83719
E-mail h.j.vanzuylen@tudelft.nl

Submitted to TRB 31 July 2008, revised 14 November 2008

Word count

Abstract	228	
Body	5467	
Figures & tables	6	x 250
Total	7195	

ABSTRACT

Video data is used more and more often to study traffic operations. However, extracting vehicle trajectories from video by using current methods is a difficult process, typically resulting in many errors. Additionally, this requires much labor to manually correct the trajectories. This paper proposes a new method to process video data from traffic operations. We do not detect a vehicle in each picture of the video separately, but instead transform the video data such that the trajectories of the vehicles (their position over time) become visible in a single image. In this single image, the trajectories can be found by detecting lines. The difference with other methods is trajectories are detected rather than vehicles. Trajectory (line) detection is more robust than vehicle (rectangle) detection, seen from the fact that applying this method around 95% of the trajectories are detected correctly and, more importantly, the length of the segments of each of the trajectories is much longer, compared to results from other methods reported in literature. Also, the detection is a very quick process since only a single image is required to be analyzed. For a dataset of 5 minutes in length, transforming costs several minutes, and automatically detecting and tracking costs 40 to 50 minutes per lane. Further manually correcting is then necessary which costs about 10 minutes per lane. In contrast, using a different method the total processing time for analyzing the traffic operations costs approximately one week for all lanes together.

INTRODUCTION

Traffic congestion forms an increasing problem in modern society. Traffic engineers try to improve the situation, but knowledge of traffic characteristics in general and at the bottleneck location are crucial. In early days, most traffic-related studies were carried out using macroscopic data. The first attempt to find the relation between density and speed was made 75 years ago by Greenshields (1). Chandler et al. (2) argued the necessity of data on individual driving behavior in 1958, using wired vehicles on a test-track for their experiments. Treiterer and Myers (3) chose a different approach and were among the first to capture traffic operations on video in 1974. In that study, the data was processed manually.

With modern techniques it is possible to gain better knowledge of traffic on this detailed level. Observations can be made of driving behavior of individual drivers. A detailed understanding of the individual driving behavior of road users can help improve traffic flow and thus reduce delays. The data of individual drivers, i.e. microscopic data, leads to insights into, for instance, headway and traffic dynamics from which, for instance, the capacity or the stability of a traffic flow can be derived. There exist different ways of collecting microscopic data. In this contribution, we focus on video data. In this paper, the word “video” indicates a consecutive series of pictures illustrating traffic operations with a fixed time-interval, regardless of the picture size and frequency. The word “picture” here always indicates a single time-frame of the video.

The time-dependent coordinates of all vehicles give a complete overview of all traffic operations. From the video, these coordinates over time are known, which gives insights in the dynamics of driving. This paper discusses the process of extracting this information from given video images. The newly proposed method gives more robust results than existing methods. This is the consequence of the fact that in other methods independent vehicles are detected, leading to difficulties in linking the vehicles found in consecutive pictures in order to get vehicle trajectories. When the program fails to recognize a vehicle in a subsequent picture, that particular vehicle needs to be indicated by the user. Obviously, these manual corrections are time-consuming. In the method proposed here, however, the duration of the parts of found trajectories are much longer, which is an advantage. Therefore, in using the proposed method, the time needed to manually complete the set of trajectories is much less. Another advantage of the newly proposed method is that it provides insight into the traffic operations without actually tracking the vehicles. That is, relevant properties of the traffic are put directly into a single image. This way, the remote sensing data are made more easily accessible.

In the next section, an overview is given of data collecting and processing methods for individual vehicles reported in literature. After this literature review, it is briefly stated how data captured from a (moving) helicopter are transformed into a series of stable images. The main section of the paper then describes how the video can be transformed. It shows which operations can be applied to combine all relevant information in a single image. The section thereafter discusses how trajectories can be detected automatically from this (transformed) image. The focus in this process is on obtaining the trajectory data. The final section then summarizes the presented research and gives concluding remarks.

LITERATURE REVIEW

This section provides an overview of how microscopic traffic information can be gathered. Several methods and the advantages and disadvantages of methods are shown first. Which

data collection methods are most appropriate and which data can be extracted is briefly discussed. This section shows that video data is a good option for several applications. Then, the second subsection discusses the methods literature provides to collect and process video data as well as the difficulties in processing video data into trajectory data.

Microscopic Traffic Data

Various studies require traffic data on the level of individual vehicles, which can be collected using a variety of methods. An exhaustive overview thereof as well as their applications can be found in Appendix C of the thesis of Ossen (4). Here, we only briefly discuss the most frequently used methods.

One of the ways of collecting microscopic data is recording the information obtained from (double) loop detectors in the road at vehicle level which provides passing times (and therefore headways) and speeds at a specific location (see e.g., (5)). However, since the cars are not followed over time, this does not give information about the dynamics of traffic.

A second way of data collecting is by GPS which can be used to record information on the speed and location of an instrumented vehicle over time. This provides the dynamics for that particular instrumented car (see e.g., (6)), but does not give insights into the interaction of vehicles. Only when there is a platoon of equipped vehicles, there is insight into the relative positions and relative speeds of the vehicles. However, this is only realized in an experiment at a test-track or in an artificially created situation on the road.

A third way of data collecting is using an instrumented vehicle. For instance, vehicles equipped with a radar can provide a more representative view on the vehicle dynamics (e.g., (7)). The radar can measure the distance between two vehicles (distance to leading vehicle, or to the following vehicle, or both). Note that the radar can be used to monitor the driving behavior of the driver of the equipped vehicle, which is called the active mode, or to monitor the behavior of the following vehicle, which is called the passive mode. In the active mode, only a few drivers can be studied, whereas in the passive mode, every driver that follows the instrumented vehicle can be studied. The disadvantage of the passive mode is that the test vehicle influences the traffic. Additionally, the distances measured by the radar can be combined with a GPS-signal to see the influences of the location on the car-following behavior. The disadvantage of using an instrumented vehicle is that the observations for different cars are made in different conditions because the instrumented vehicle itself is moving. That means that either all observations are made at different locations, or, in case the instrumented vehicle returns to the same location, then observations are made at different time instances and under different traffic conditions.

A fourth way of data collecting is capturing video from a high point of view. This provides insights into the driving behavior of drivers both longitudinally (e.g., car-following), and laterally (e.g., lane selection). Consequently, one can analyze the headway choice, like using an instrumented vehicle, but also, for instance, the influence of the distance to the leader's leader on the acceleration. So, one advantage is that the information has both a spatial and temporal dimension. Another advantage is that many drivers can be observed under similar driving conditions.

Collecting and Processing Video Data

It is only a few years ago that projects started using remote sensing data on a larger scale, for instance Tracing Congestion Dynamics at the Delft University of Technology or The NGSIM (Next Generation Simulation) project (8) which provides data of individual vehicles. These are used in many projects in the US, for a wide variety of applications, from analyzing signalized intersections (9) to lane-change behavior (10) and emergency behavior (11). Also

in Europe, these US data are used in several projects, for instance to show properties of the fundamental diagram (12) or to calibrate car-following models (13).

Before these video data can be used, these need to be processed. Hoogendoorn et al. (14) propose a method of first finding vehicles by comparing pixel values in a picture with the median value of that pixel (i.e., that position) over time, which can be interpreted as the brightness of the roadway background. Then, clusters of pixels deviating from the values identified for the background are marked as vehicles. This process is repeated for each picture. In the final step, the vehicles are tracked by comparing the location of vehicles in two subsequent images. A detection of over 90% is claimed.

However, this method has shown to have some flaws when we applied it to a dataset with different shades of tarmac and different light intensities. In practice, it turned out that the trajectory of a vehicle was interrupted over time. This makes it necessary to combine partial short trajectories. A normal filter procedure would discard too short trajectories, which could be interpreted as noise. Because the trajectories of vehicles were interrupted and short, such a filter procedure could not be applied now.. Additionally, other objects than vehicles were detected (e.g., road lines), which we could not simply ignore. This gave some serious trouble regarding the combination of detected parts since it was unclear whether the parts were vehicles or other objects. In short, the main problem was the tracking.

A similar methodology is stated by Angel et al. (15). In each picture of video taken from a helicopter, the intensity is compared with a predefined pixel value. A concentration of deviating pixels is marked as a vehicle. In contrast to Hoogendoorn et al (14) where only the information on the pictures is used to find reference points, Angel et al. (15) use added information collected during the flight: the position (GPS-coordinates) and the angle of the helicopter is saved together with the video data. The main problem indicated here is the "matches" of vehicles, i.e. to connect the same detected vehicle over different frames. They also report a matching percentage of approximately 90%.

In the NGSIM-project (16) a subproject deals with extracting the trajectory information from the video, NGSIM-video (17). There, vehicles are found in each image, but also origins and destinations are to be provided. In the tracking process, this information on flow direction is used. The tool has a possibility to track both forward and backward in time. No reliability value is given for the automated procedure.

However, given the similar methodology to both (14) and (15), it is likely that similar problems occur. Additionally, the manual states that there is a need to post-process the tracks manually, for which a tool is provided. This is time-consuming work: 'Experienced NGSIM-VIDEO operators can effectively process video at a rate of 40 – 100 seconds of tracking per work day for a 500-meter freeway section.'(17).

Cho and Rice (18) present a method to estimate the velocity field from video data, based on the image intensity in each lane. To find a vehicle, they take the maximum value for the image intensity across a single lane. This method detects bright vehicles, but darker vehicles cannot be identified since the maximum intensity value on a cross section is the intensity of the pavement. However, this is not a problem for determining the velocity field, which can be derived from the bright vehicles only. For the same reason, unmatched parts of a trajectory do not form a problem, since it is sufficient to use the parts that are available. However, a more detailed analysis requires the full trajectories of all vehicles.

There is therefore need for a method which gives the trajectories of vehicles from remote-sensing data reliably for longer time intervals without much manual post-processing.

TRANSFORMATIONS

This paper proposes to reduce the problem with one dimension and to capture all relevant information in one image. This section explains which transformations can be made in order to construct these images.

The video need to be stabilized, if captured from a moving helicopter, and rectified, if. if the viewing angle is not perpendicular to the road surface. Due to space restrictions, we will not discuss this process in this paper. For this paper, the starting point is a series of stabilized and rectified pictures. Usually, vehicles are identified in each picture, and this process is repeated for each picture. Note that we will use the word picture for a time slice of the video (the snap-shot of the traffic conditions in one moment in time), whereas the word image is used for graphical representations which are constructed otherwise.

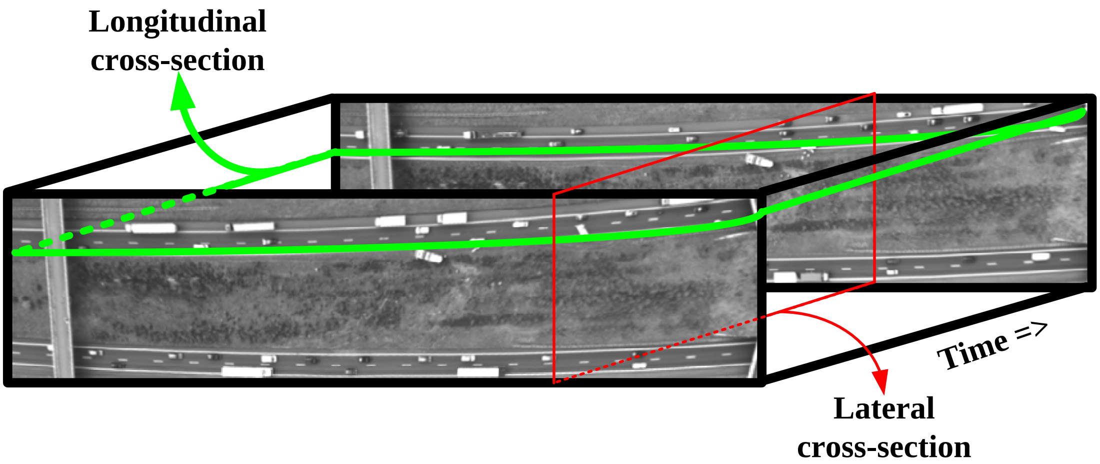

Figure 1 A three-dimensional box of pictures with the lateral (red) and longitudinal (green) cross-section

The time series of pictures can be combined to a three-dimensional box, with two space dimensions and a time dimension (Figure 1). It is proposed to take a cross-section (i.e., a slice or cross-cut of this box) in one of the space dimensions. The first subsection shows the transformation in the lateral direction, perpendicular to the driving direction, indicated by a red frame in Figure 1. For one point along the road, the image shows the occupancies over time and the flows. The second subsection shows the transformations in the longitudinal direction, along the driving direction, indicated by a green frame in Figure 1. This transformation gives the occupancy over space and time for one lane, directly indicating the vehicle trajectories.

The video used here as example and to quote computation times originate from a experiment described in (19). The pictures with a size of 1392x1040 pixels are recorded at a frequency of 15.1 Hz.

Lateral cross-section

Figure 2 shows a lateral cross-section of a series of pictures, or, expressed differently, it is a sectional plane perpendicular to the driving direction, like the red plane in Figure 1. Note that the size of the image is a distance, 50 meters from bottom to top, times a time, 75 seconds from left to right The solid lines indicate a time step of half a minute, the dotted lines are indicate a time step of 15 seconds.

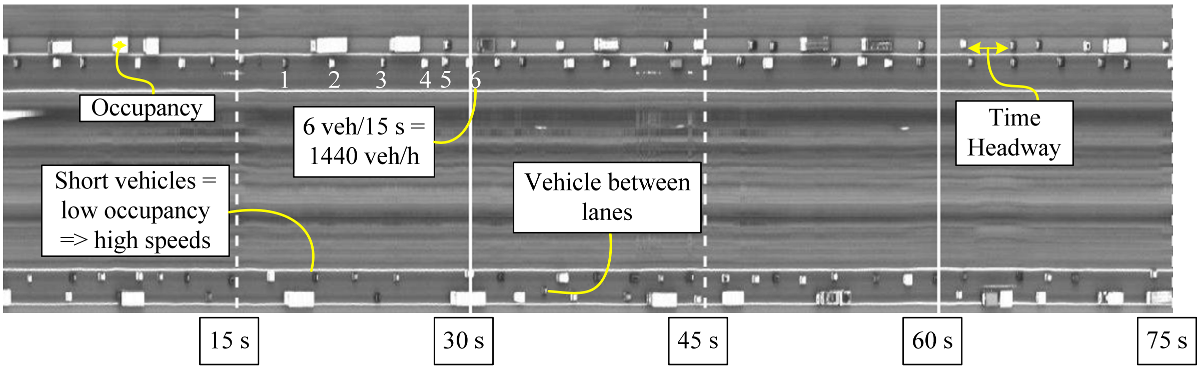

Figure 2 A lateral cross section, size 1.25 min x 50 m

This image can be interpreted as the occupancy of a virtual loop-detector over time. It essentially shows when, at the horizontal axis, a vehicle passes a point along the road. Data from detection loops can lead to a similar image. However, this image also has a space component. Therefore, the image shows where on the lateral y-position a vehicle passes the point of the cross-section. This image can be constructed in a few minutes' time.

The image shown here is easily confused with a normal picture of traffic. This is because the structure of the road seems well visible. However, the length of the vehicles at the horizontal axis varies strongly since in this image the length of a vehicle is the time it would occupy the virtual detector and therefore faster driving cars are drawn shorter,

In this image, cars in both directions have the nose at the left and the back at the right. Each vehicle first reaches the cross section location with the nose of the vehicle. Since left in the image is an early point in time, the nose of the vehicle is drawn at the left hand side. This is most easily recognized by articulated lorries where the truck is at the left and the lorry at the right.

The transformation proposed in this subsection is ideal for getting time headways and flow values out of remote sensing data. The time headway can be read out as the distance between two passing cars as indicated in the figure. To obtain the flow, it is required to count the number of vehicles that pass by in a time interval, either manually or automatically, which is preferred for a longer dataset.

This type of images is very informative for the lateral position on the road. It can directly be seen whether cars within a platoon drive at the same lateral position or whether that depends on their position in the platoon or the headway or the type of car they are following. It furthermore can be derived whether the headway depends on the type of car (passenger car, truck) drivers are following.

There is one pitfall. Vehicles are not detected if they are upstream of the virtual detector at one picture and they are downstream of the detector in the next. However, when recoding pictures at a time interval of (in our case) 15 images per second or more, that is very unlikely. In the worst case, there is a small (4 meter = 13 ft) vehicle that just does not reach the cross section point when the first image is taken. When the next picture is taken, it should have passed the point completely. So it should have covered 4 meters in 1/15th of a second, and thus should drive 60 m/s, which equals over 200 km/h or over 120 mph. For a frame rate of 10 images/second, the critical speeds is 145 km/h (90 mph). For longer cars these speeds are proportionally higher.

From this image, one cannot derive the dynamics of the car-following behavior. This can be derived from a longitudinal cross section discussed in the next subsection.

Longitudinal cross-section

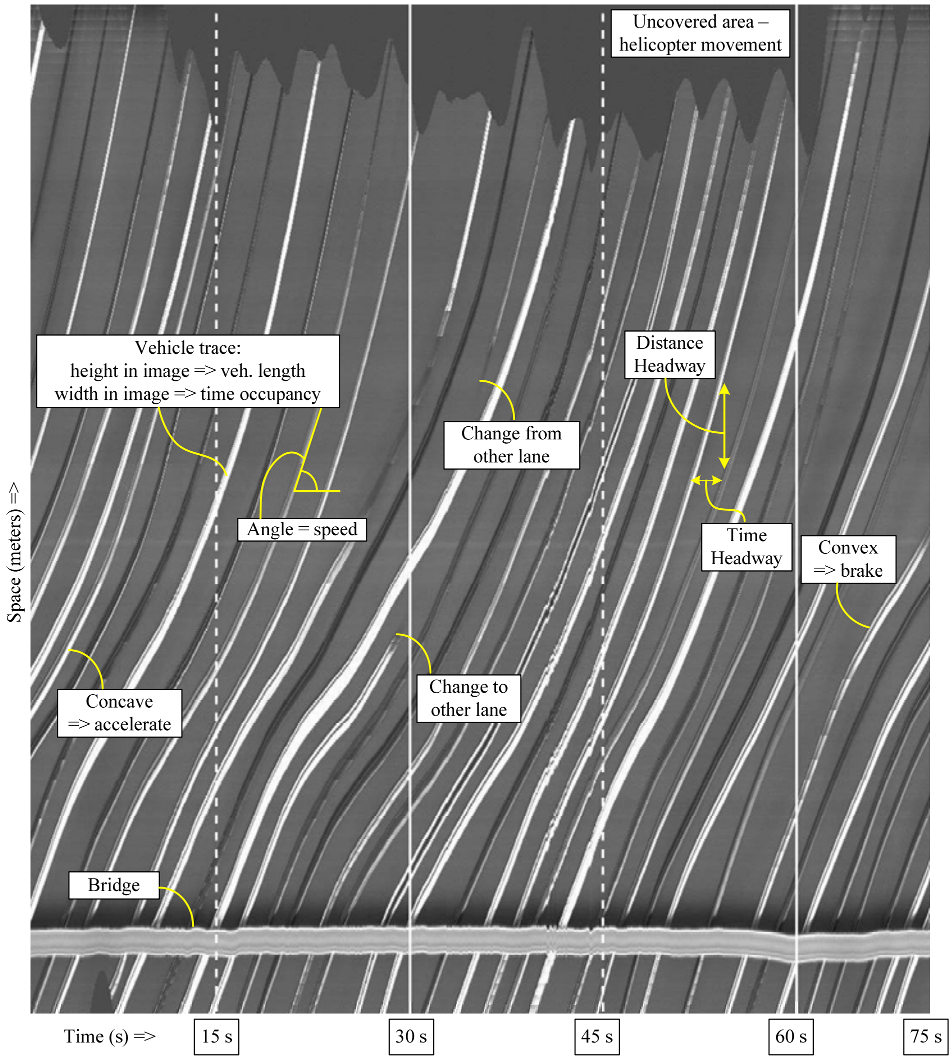

Figure 3 A longitudinal cross-section

Figure 3 shows a lateral cross-section image along the axis of the lane of a road, like indicated with a green frame in Figure 1. On the horizontal axis, the time is plotted which progresses from left to right). Again, the vertical lines indicate 15 seconds (dotted lines) and 30 seconds. On the vertical axis, one finds the distance along the road with the traffic flowing from bottom to top. The cross-section is not necessarily taken along a straight line. A curved road requires that one takes a section along a curved line, as is shown in Figure 1.

Cho en Rice (18) already propose a method filtering one lane out of a set of pictures. However, they take one *lane* out which is several tens of pixels wide instead of one pixel row as we proposed. Then the maximum of the pixels in that lane across the road is taken, along a lateral cross-section of the lane. This gives a clear image of the bright cars, but the dark cars disappear since the maximum intensity value of the pavement is higher than those of a dark car. We propose to directly narrow the data to a one-pixel wide cross-section. This is, to the best of the authors' knowledge, the first time such an approach is taken in the field of traffic flow.

Figure 3 is constructed from the video data in a few minutes. It shows the intensity values of the (grayscale) images. The background of the image is the gray shade of the pavement. Vehicles move from the bottom (downstream) part of the image to the upper part. A bright line indicates a bright vehicle moving, a dark line the passing of a dark vehicle. This

is a graphical representation of the trajectory that is indicated by the word “trace” in this paper. The length of the vehicle determines the width of the line in this image, the occupancy time at one location determines the height of the line.

At the bottom part, a bridge blocks the view on the road which is (obviously) present during the whole observation. It therefore forms a line from left to right. Because the bridge is higher than the road surface, the part of the road that is made invisible by the bridge changes slightly when the helicopter moves and therefore the line is not straight.

The angle that the trace of a vehicle makes with the horizontal line is its speed. In this way, the image shows the speeds of all cars at all times. Also the changes of speeds are visible since concave traces equal a reduction of speed and therefore braking, whereas convex traces equal an increase of speed and therefore accelerating.

Lane changes are represented by an appearance or disappearance of the trace of a vehicle in Figure 3 of which examples are indicated in Figure 3. In this example, the following vehicle accelerates and closes the gap to the predecessor. Another vehicle subsequently merges into the gap that is now created behind the accelerating vehicle. The lane-changing vehicle enters this stream and thus appears in the image.

Using this longitudinal cross-section image, the dynamics of the traffic operations can be made visible without advanced detection and tracking algorithms. The image is constructed in typically several minutes one 1.5 GHZ core of a dual core pc. The constructing time is only determined by the speed of reading the images because here are no advanced image processing algorithms required. Note that in this way in a few minutes the dynamics of vehicle movements are made visible without fitting a model.

Contrary to the lateral transformation, this longitudinal transformation provides no information of the lateral position of the vehicles, other than the lane in which it is driving. In most cases, however, the only lateral information that is needed is the lane in which a vehicle is driving. Apart from that, each lane is shown in a different image. Nevertheless, one can track vehicles that change lane by combining the tracks of the vehicles in different lanes.

VEHICLE DETECTION

This section describes how vehicles can be detected and tracked in the cross-sectional images. As example we take the longitudinal cross-section image, because the trajectory information is the information that is more often required in studies using remote sensing data. The method of finding the cars (described in subsection threshold values) is applicable for the lateral cross-section image as well. However, the smoothing, described in the subsequent section, is only needed for the trajectories found in the longitudinal cross-section images. In practice it turned out that if we use a longitudinal cross-section image we can track a single vehicle for a longer time than if we use procedure which detects and tracks vehicles in each picture separately. Also the manual corrections are much easier than in the existing methods. Both the automatic part and the manual corrections are much quicker than the methods proposed so far. Namely, an image of a dataset of 5 minutes in length can be processed (detecting and tracking) in 40-50 minutes. The manual corrections cost around 10 minutes.

The images themselves provide a lot of information on traffic flows which can easily extracted manually. The sequel of this section describes how this can be done automatically. However, fitting a polynomial function through the lines of a longitudinal cross section image gives the trajectories very accurately. In fact, the extent to which the polynomial curve follows the trace can be influenced by the number of points one uses to fit the trace. Generally, the number of points will be higher for vehicles with more fluctuation in speed. Therefore, choosing for either an automatic detection or a manual identification is a trade-off between

programming time and optimization of the algorithm at one hand and working time (clicking points in the image) at the other hand.

Threshold values

The images need to be changed from a gray value to a Boolean value (true or false) indicating vehicle presence. Compared to existing approaches, e.g. (14, 15), we make two changes in the process of detecting vehicles. First, we do no take the values of one pixel over the whole time as basis for the threshold values because this gives some problems with the change of light intensity due to clouds passing by. It works much better if a shorter time (10 seconds) is considered. However, this also has a drawback, since, for example, a truck could be at the same spot for more than half of the time. In that case, using a median for the background, the intensity value of the truck would actually be considered to be the background and therefore the pavement would be seen as vehicle. To avoid this problem, we clustered the pixels in blocks of around 50 meters (around 150 pixels), which gave many more pixels to extract the distribution of intensities. This gives "blocks" of around (10 seconds =) 150 pixels times (50 meters =) 150 pixels to find the background intensity. The second difference with existing methods (e.g.,(14)) is that we did not determine the background and found a deviation of it. In contrast, with upper and lower percentile values are used which turned out to give more reliable results.

For each of the "blocks" mentioned in the previous paragraph a lower and upper value are computed. This can be seen in Figure 4 where the intensity along one line is plotted. Note, the background intensity profile varies along the road which for instance can be caused by the different reflection of the road at various angles. The threshold values for each of the blocks are interpolated linearly to obtain the threshold values for each pixel individually. This pattern is also plotted in Figure 4.

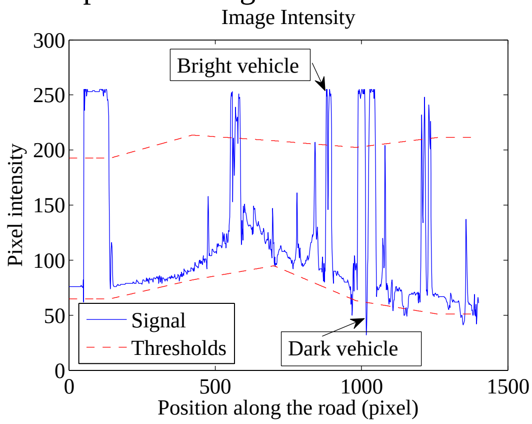

Figure 4The intensity of the image over space.

Note that the percentage of pixels marked as car is fixed by choosing the percentile values. Moreover, it also determines whether the dark or the light deviations will be marked as vehicle. Therefore, the choice for the best percentile values depends, among others, on the traffic conditions and the brightness of the pavement relative to the brightness of the vehicles. For example, in case there is much traffic, the lower percentile value should be chosen low and the high percentile value should be chosen high to assure a large part of the pixels is marked as vehicles. And in case the pavement is light, the majority of the vehicles will be darker than the pavement and therefore both the lower and the higher percentile values can be increased. In practice it is best to try different settings of the percentile values. In our case, it was best to choose 15% for the lower percentile value and 90% for the higher percentile value.

Special attention should be given to shadows since they might interfere with the vehicle detection. For instance, if the shadows lie over the adjacent lane, it is advisable to choose a pixel line of the video more to the side of the lane where no the shadows lie. This was possible for our test dataset. If it is unavoidable to take a pixel line with shadows, the percentile values have to be changed. In particular, the lower percentile value has to be increased to avoid that the shadows are detected as cars. Similarly, shadows could be detected as part of a car if they lie along the road which could increase length of the area being detected as different from the background and therefore increase the vehicle length and thus the inter-vehicle spacing. To solve this, the lower threshold value should be adjusted.

Another problem arises when there are vehicles with a color which gives a pixel intensity similar to or equal to the pavement. Vehicles with the exact same reflection as the pavement cannot be detected. However, in our test case there were no such vehicles, as could be derived from the shadows. Let us finally remark that distinguishing vehicles from the pavement is easier if color video is used.

Vehicle presence now has to be filtered. First, the image with traces (in the space-time plane of the longitudinal cross-section) is filtered by applying an exponential moving average filter with an exponent of 2 pixels (2/15 of a second) in the time direction and 1 pixel (30 cm) in the space dimension to filter out the small areas that deviate from the neighboring pixels. The result is a smoother image. Then the image is filtered based on the properties of cars. Any vehicle has to have a length of 3 (10 feet) to 20 meters (66 feet). If the vehicle is too small, the signal is ignored because this is noise. Additionally, a gap between two vehicles has a minimum size of 3 meters (10 feet). If the gap is too small, the two parts are combined since this is either noise or for instance an articulated lorry.

Now, the vehicles found in each time step should be matched to create trajectories. In two subsequent lines, the expected displacement of the vehicle can be calculated based on speed; in case there is no speed known, for instance in the first time step, a fixed estimate can be used. If the match of size and location is better than a threshold value, the detected vehicles are matched to make a trajectory.

This will give the trajectory in some distinct parts. In some time instants namely vehicles are not detected properly which means there are many “partial” trajectories which have to be coupled in the next step. Of trajectories which end at another location than at the edge of the image have, a majority has to be coupled to another partial trajectory; only if the vehicle changes lane it disappears from the image. The coupling of this partial trajectory works as follows. Each partial trajectory is extrapolated, based on the position and speed slightly before it ends. In this process, it is better not to use the very last detected position since the measurement gets noisy at the last measurement before it is not detected anymore. This extrapolated trajectory is compared with every starting partial trajectory. If one of them matches in space and speed, both partial trajectories are coupled. The maximum allowed error in this matching process can be chosen. If we choose this maximum high, more partial trajectories are coupled, but this increases the probability of coupling partial trajectories which do not belong to the same vehicle, whereas a lower threshold leaves some partial trajectories uncoupled, but there are fewer errors in the coupling. We choose for a relatively strict option.

This choice means that several partial trajectories remain uncoupled: on average, each vehicle trajectory now consists of 2 partial trajectories. This is now corrected manually. To this end, the partial trajectories and a unique number identifying the trajectory are plotted (Figure 5). In an Excel sheet we now indicate which partial trajectories should be combined. This takes around 10 minutes for a video of 5 minutes in length. From this point, the further processing is automatic.

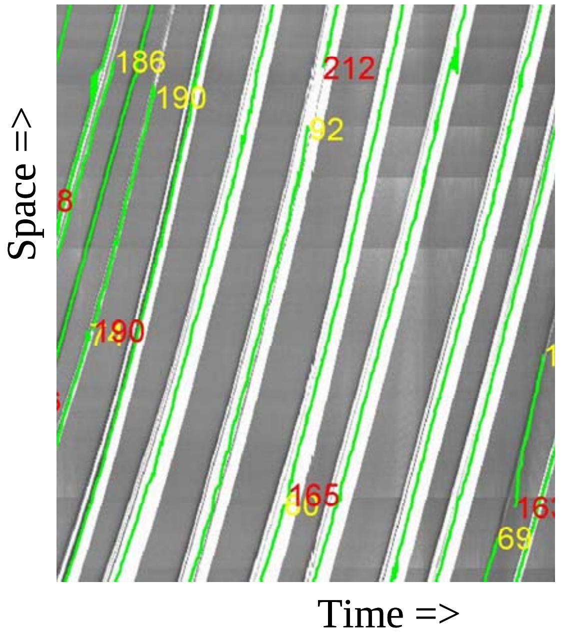

Figure 5 The trajectories found on top of the image of the longitudinal cross-section.

Smoothing

The procedure described in the last subsections gives us for each vehicle the trajectory. Nevertheless, these trajectories are not smooth yet, as Figure 5 shows. This is mainly caused by the spatial and temporal resolution of the images, but also by detecting and tracking errors. To represent the reality, they need to be smoothed, without smoothing out the dynamics of traffic, To this end, the method of Toledo (20) is applied. This method averages the position at a time step with the position at neighboring time steps, with a larger weight for time steps which are closer by. The smoothing filter takes time frames into account with times that differ at maximum 1.5 seconds from the current time frame. An example of the end result, the trajectories, can be found in Figure 6.

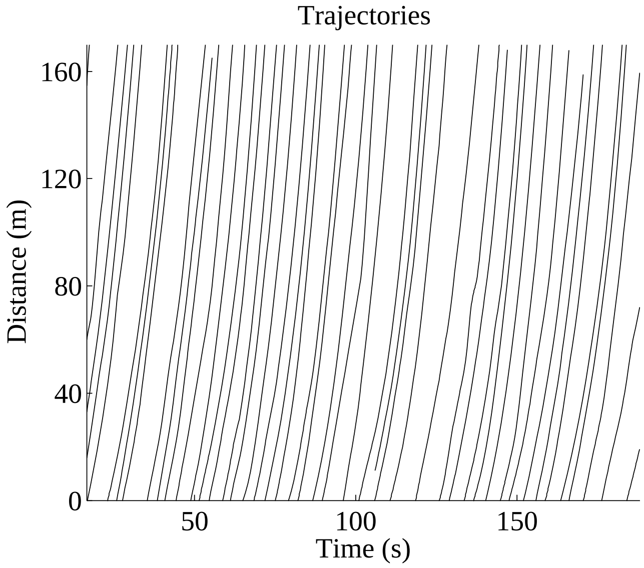

Figure 6Trajectories for several vehicles

CONCLUSIONS

Video data from traffic operations contain much valuable information. However, extracting, for instance trajectory data using existing methods is a difficult and often erroneous process. The approach proposed here increases the length of the parts of the trajectories, thereby overcoming the shortcomings of existing methods. To this end, instead of analyzing each picture in a video sequence separately, the data of the video is transformed in such a way that the vehicle trajectories (vehicle positions over time) become visible within one single image.

For instance, the cross-section of the three-dimensional box along the axis of the road provides insight in a quick and easy manner into the dynamic car-following behavior. Within a few minutes, the traces of the cars, i.e. the vehicle trajectories, are plotted in a space-time diagram. Then, from the constructed image, many dynamic properties of the traffic are apparent, such as shockwaves, reaction times, and driver over-reaction. Using image detection techniques, the vehicle positions can be quantified. This is a very reliable method of constructing trajectories.

In other methods, the vehicle trajectories are often divided into many different segments. In the presented method, the length of the segments can be longer, as well as the segments are more easily coupled. Close to 95% of the vehicle presence (in the space-time frame) is detected, while the remaining 5% can be manually reconstructed very quickly. The complete process consisting of transforming, detecting, tracking, and manually correcting of one single lane can be performed in approximately 1 hour for a 5-minute dataset using a PC equipped with 1.5 GHz core processor.

Another cross-section of the three-dimensional box, perpendicular to the driving direction, shows other traffic properties. Headways and occupancies can namely be derived directly from this single image. For these purposes, it is not necessary to detect all vehicles in all pictures, but only the vehicles presently passing the location of the virtual detector.

In our analyses, black and white images were used. For the case of using color images, the contrast between vehicles and pavement increases, making it easier to detect these vehicles. The tracking process also benefits from color since vehicles detected in consecutive pictures can then be identified and linked by color. Thus, in the future, the use of color images is expected to further increase the length of the automatically discerned vehicle trajectories.

ACKNOWLEDGEMENT

This research was supported by the research program Next Generation Infrastructures. The authors also thank the reviewers for their helpful comments which improved the paper.

REFERENCES

1. Greenshields, B.D. A Study of Traffic Capacity. In *Proceedings Highway Research Board*, Vol.14, 1934, pp. 448-477.
2. Chandler, R.E., R. Herman, and E.W. Montroll. Traffic dynamics: studies in car following. In *Operations Research*, Vol.6, 1958, pp. 165-184.
3. Treiterer, J. and J.A. Myers, eds. *The hysteresis phenomenon in traffic flow*. Proceedings of the 6th International Symposium on Transportation and Traffic Theory. 1974, Elsevier. 13-38.
4. Ossen, S.J.L., *Longitudinal Driving Behavior: Theory and Empirics*. 2008, Delft University of Technology: Delft.
5. Hoogendoorn, S.P. Unified approach to estimating free speed distributions. In *Transportation Research Part B: Methodological*, Vol.39(8), 2005, pp. 709-727.

6. Brockfeld, E., R.D. Kühne, and P. Wagner. Calibration and Validation of Microscopic Traffic Flow Models. In *Transportation Research Record: Journal of the Transportation Research Board No.1876*, TRB, National Research Council, Washington, 2004, pp. 62 - 70.
7. Brackstone, M., B. Sultan, and M. McDonald. Motorway driver behaviour: studies on car following. In *Transportation Research Part F: Traffic Psychology and Behaviour*, Vol.5(1), 2002, pp. 31-46.
8. *Next Generation SIMulation Fact Sheet*, in *NGSIM Overview*. Publication FHWA-HRT-06-135. Federal Highway Administration, 2006.
9. Kyte, M., M. Dixon, T. Urbanik, V. Nayak, and A. Abdel-Rahim. What's New in the Queue: Discovering More about Queue Discharge Characteristics and Their Effect on Signal Timing Using the New NGSIM Data Set. In *Proceedings of the 87th Annual Meeting of the Transportation Research Board*, 2008.
10. Choudhury, C., M.E. Ben Akiva, T. Toledo, A. Rao, and G. Lee, *Cooperative Lance changing and Forced Merging Model: NGSIM Final Report*. Federal Highway Administration, 2006.
11. Hamdar, S.H. and H.S. Mahmassani. From Existing Accident-Free Car-Following Models to Colliding Vehicles: Exploration and Assessment. In *Proceedings of the 87th Annual Meeting of the Transportation Research Board*, Washington D.C., 2008.
12. Chiabaut, N., C. Buisson, and L. Leclercq. Fundamental Diagram Estimation through Passing Rate Measurements in Congestion. In *Proceedings of 87th Annual Meeting of the Transportation Research Board*, Washington D.C., 2008.
13. Kesting, A. and M. Treiber. Calibrating Car-Following Models using Trajectory Data: A Methodological Study. In *Proceedings of the 87th Annual Meeting of the Transportation Research Board*, Washington D.C., 2008.
14. Hoogendoorn, S., H.J. van Zuylen, M. Schreuder, B.G.H. Gorte, and M. Vosselman. Microscopic traffic data collection by remote sensing. In *Transportation Research Record: Journal of the Transportation Research Board No.1885*, TRB, National Research Council, Washington D.C., 2003, pp. 121 - 128.
15. Angel, A., M. Hickman, and P. Mirchandani. Methods of Analyzing Traffic Imagery Collected From Aerial Platforms. In *IEEE Transactions on Intelligent Transportation Systems*, Vol.4(2), 2003, pp. 99 - 107.
16. www.ngsim.fhwa.dot.gov/. [cited 2008 29 July 2008].
17. Zhang, L., V. Kovvali, N. Clark, Douglas Sallman, and V. Alexiadis, *NGSIM-VIDEO User's Manual*. Publication #FHWA-HOP-07-009. US Department of Transportation, Federal Highway Administration, 2007.
18. Cho, Y. and J. Rice. Estimating Velocity Fields on a Freeway from Low-Resolution Videos. In *IEEE Transactions on intelligent transportation systems* Institute of Electrical and Electronics Engineers,, Piscataway, NJ, USA Vol.7(4), 2006, pp. 463-469.
19. Knoop, V.L., S.P. Hoogendoorn, and H.J. van Zuylen. Capacity Reduction at Incidents: Empirical Data Collected from a Helicopter. In *Proceedings of 87th Annual Meeting of the Transportation Research Board*, Washington D.C., 2008.
20. Toledo, T., H.N. Koutsopoulos, and K.I. Ahmed. Estimation of Vehicle Trajectories with Locally Weighted Regression. In *Proceedings of the 86th Annual Meeting of the Transportation Research Board*, Washington D.C., 2007.