# Vehicle Speed Measurement Using Stereo Camera Pair

Pavel Najman and Pavel Zemčík, *Member, IEEE*

***Abstract*— We have proposed a novel method for vehicle speed estimation using a calibrated and synchronized pair of stereo cameras. In a newly proposed method, we first localize the vehicle by detecting and tracking its license plate in a series of stereo images; then, we triangulate the vehicle position along its trajectory; and finally, we compute its speed based on the trajectory and time. The experiments show that the proposed method overcomes state-of-the-art results with a mean error of approximately 0.05 km/h, a standard deviation of less than 0.20 km/h, and a maximum absolute error of less than 0.75 km/h. For the purpose of evaluation, we have recorded a dataset that contains over 600 vehicles whose trajectories were recorded and for which their ground truth speed was obtained from a pair of single beam LIDARs in optical gate configuration. Using the presented method, the speed was measured for over 99 % of the recorded vehicles. Others were rejected by the method mainly due to their short trajectories, obstructed license plates or frame errors that would adversely affect the precision of the measurement.**

***Index Terms*— Vehicle speed measurement, stereo matching, sub-pixel registration, stereo vision.**

## I. Introduction

THE speed of a moving vehicle directly influences both the risk of a crash and its consequences. To minimize this risk and to increase the road traffic safety, speed limits are imposed that should ensure that in the event of a crash, impact energies remain below the threshold likely to produce either death or serious injury.

The threshold usually depends on the most probable crash scenario which varies with road location. In residential and high pedestrian traffic areas it is usually around 30 km/h. In cities or in areas with a higher probability of side impact of vehicles and with a lower amount of pedestrian traffic, it is around 50 km/h. And for highways, where rear-end collisions are prevalent it is around 130 km/h [1].

In order to enforce the imposed speed limits, individual countries typically implement metrological legislation that defines possible tolerances of measurement devices usable for speed enforcement. In most cases, the maximum allowable tolerance is in the units of percents, in the case of the EU, for example, the maximum allowable tolerance is 3 % of the measurement device range (i.e. in the case of 100 km/h range, the maximum allowable tolerance is $\pm 3$ km/h). Moreover, tolerance should be met with a 99.9 % probability in this case (in the metrological evaluation procedure, only one in 1000 measurements can produce an erroneous result). Therefore, if the distribution is Normal, the standard deviation should not exceed 1 km/h.

The tolerances vary by country. Some countries leave the enforcement tolerance up to the discretion of the arresting officer or, in case of automatic systems, on the device manufacturer. Other countries have an official tolerance specified by law. The official tolerance places a requirement of a maximum allowed measurement error on a speed measuring device. Consequently, only those devices that fulfil the maximum error condition can receive proper certification and can be used by officials for speed limit enforcement. The tolerances range from very strict - 1 km/h (e. g. in Sweden) - to very lenient 20 km/h (e. g. in Russia) [2].

The speed measurement devices are usually classified as either intrusive or non-intrusive [3]. Pneumatic tube detectors, inductive loops, magnetometers and piezoelectric sensors belong to the intrusive category. Although the devices are accurate, and by themselves low cost, they need to be embedded into the road. Their installation and maintenance are therefore problematic and expensive because they usually require lane closure, which disrupts the traffic, and a pavement cut with a subsequent repair or resurfacing of the road.

The non-intrusive devices are placed either above the road or by its side which makes their installation and maintenance easier and cheaper in comparison with intrusive technologies. Ultrasonic sensors, infrared sensors, radars and camera-based technologies fall into this category. Ultrasonic, infrared and radar systems all transmit a signal and analyze the signal that was received based on the transmitted one. The speed can be measured either indirectly, e. g. by detecting the vehicle at the start and the end of the detection zone, or directly, e. g. by exploiting the Doppler effect [4], [5]. Nowadays, more attention is given to camera-based technologies because they provide a rich array of data that can be quickly processed on contemporary hardware. Single camera devices are very interesting from an application perspective and a lot of work has been done to improve the methods they use for calibration and speed estimation. But certification of these devices is problematic. Stereo camera devices, on the other hand, are based on more transparent calibration and speed estimation methods and their certification should be easier. In our case,

Manuscript received March 20, 2020; revised August 7, 2020; accepted October 6, 2020. Date of publication November 10, 2020; date of current version March 9, 2022. This work was supported by the Ministry of Education, Youth and Sports from the National Programme of Sustainability (NPU II) Project IT4Innovations Excellence in Science under Grant LQ1602. The Associate Editor for this article was S. S. Nedevschi. *(Corresponding author: Pavel Najman.)*

The authors are with the Center of Excellence IT4Innovations, Faculty of Information Technology, Brno University of Technology, 612 00 Brno, Czech Republic (e-mail: inajman@fit.vutbr.cz).

Digital Object Identifier 10.1109/TITS.2020.3035262

the goal is to provide easy and metrologically acceptable calibration and measurement algorithms that would be feasible to implement in real-life devices.

Three main stereo reconstruction approaches include local [6], semi-global [7], and global [8]. In our case, we need only several individual points along the vehicle trajectory to compute its speed; therefore, the reconstruction of the whole scene is not necessary. The approach we use to get the trajectory could be considered local and sparse.

The key contributions of this paper are:

- Localization of the vehicle by detecting and tracking its license plate in a series of stereo images.
- Triangulation of the vehicle position along its trajectory.
- Speed computation that involves vehicle motion model.

Together they form a novel method for vehicle speed measurement using a stereo camera pair whose results comply with the EU metrological standards.

## II. Related Work

Stereovision-based methods for vehicle speed estimation usually assume synchronized cameras previously calibrated using already established methods. Their most important parts deal with feature point selection and correspondence search. Only a few papers that use a stereo camera pair for vehicle speed estimation exist.

Jalalat *et al.* [9] use a vertical stereo setup pre-calibrated using a chessboard pattern [10]. After background subtraction in a selected ROI, they detect and track vehicles using a Viola-Jones cascade classifier and Kalman filter. The feature points are selected by uniform sampling in the lower part of a detection bounding box and corresponding points are found with sub-pixel precision by exploiting the single-step DFT technique. The average vehicle speed is expressed in terms of distance travelled, computed from triangulated vehicle positions, per time between two frames. They report the speed measurement error as an arithmetic mean of absolute error percentages compared to reference measurements by Fama Laser III. The worst mean percentage error was 3.3 %. The absolute error was not reported.

El Bouziady *et al.* [11] use a horizontal stereo laboratory pre-calibrated setup. After background subtraction, vehicles are detected and tracked as convex blobs. The SURF detector and descriptor is used to select the important points on the vehicle and to find the point correspondences. They computed the average speed in the same fashion as Jalalat *et al.* [9]. They compared the measured speed with ground truth obtained from a GPS. The result was a mean squared error of 1.67 km/h on a dataset with a speed range of 60-90 km/h and 2.33 km/h on a dataset with a speed range of 90-120 km/h. The maximum absolute error was 2 km/h common for both datasets.

Yang *et al.* [12] use a horizontal stereo setup calibrated using Zhang's method [10]. They detect license plates using a single shot multibox detector. License plate tracking and feature point extraction and matching are done using speed up robust features (SURF) [13]. They retain only those feature point pairs that lie near the centre of the detected license plate. These points are then triangulated and their distance to the camera on the left is computed. The triangulated points whose absolute distance z-score is greater than one are filtered. Of the remaining points, the one that is closest to the centre of the license plate is considered as the exact spatial location of the target vehicle in the current stereo frame pair. Using the spatial locations of the vehicle in two frames they compute the average speed in the same fashion as in previous works [9], [11]. They compared the measured speed with the ground truth obtained from a professional satellite speed meter. Their dataset contained 4 vehicle passes with a speed range of between 20 and 50 km/h. The mean error was 0.02 km/h, mean squared error was 0.42 km/h, the maximum absolute error was 1.6 km/h with a maximum percentage error of 3.8 %.

Halfway between the stereo camera and single camera setups are two-camera setups. Llorca *et al.* [14] use two synchronized cameras with different focal lengths and non-overlapping views. Both cameras are calibrated so their intrinsic parameters and their extrinsic relationship with respect to the road are known. The license plate is used for vehicle detection and tracking is done using optical character recognition. The calibration results and the known license plate dimensions are used to compute the vehicle position with respect to the road reference. The average speed is obtained from computed positions and their timestamps. They compared the measured speed with ground truth obtained from DGPS for 8 vehicle runs at 8 different speeds (10 - 80 km/h), that is, 64 runs in total. The mean absolute error was 1.44 km/h and the maximum error was 2.62 km/h.

Most work in camera-based vehicle speed estimation focuses on single camera setups. The single camera methods are based on estimating the vehicle motion plane and scene scale. Once these two pieces of information are known, the distance that the vehicle travelled between the two frames can be retrieved by measuring the distance between two points on a rectified and scaled motion plane. The speed measurement is then simply obtained by dividing the distance by the time difference between the two frames. Different approaches on how to estimate the motion plane and scene scale exist.

A fully automatic method was presented by Dubska *et al.* [15]. This method recovers the motion plane from two originally orthogonal directions. The first of the directions is obtained from the direction of the traffic and the second one from the direction of vehicle edges perpendicular to the vehicle motion. The scene scale is acquired by constructing 3D bounding boxes around vehicles and relating the pixel dimensions to the dimensions of an average vehicle based on the statistical data of sold cars. They achieve a mean error of 1.9 % (cca 1.5 km/h) and the worst error of 4.3 %.

Sochor *et al.* [16], [17] later enhanced the method of Dubska *et al.* [15] by a fine-grained classification of vehicles, thus improving the accuracy of the scene scale estimate. They report a mean percentage error of 1.4 % and mean absolute error of 1.1 km/h. 99 % of measurements had an absolute error below 3 km/h and the percentage error below 4.1 %.

Luvizon *et al.* [18] proposed a method that requires manual input in the form of measured distances between four points on the road plane for each lane. These four points and their distances then provide a scene scale as well as a way to rectify

the road plane through homography. The travelled distance is measured as a distance between license plate feature points and the rectified and scaled road plane. But since the license plate is located above the road plane the measured speed is higher than the actual one. To mitigate that, they use a simple multiplicative constant. They achieve the mean error of -0.5 km/h with a standard deviation of 1.4 km/h. The worst errors were -4.7 km/h and 6 km/h.

Famouri *et al.* [19] refined the approach of Luvizon *et al.* [18] by measuring the distances on actual vehicle motion plane by calculating the height of the license plate above the road plane using the Shape-from-template method [20]. By doing that, they reduced the mean error of Luvizon *et al.*'s [18] method by 25 %.

## III. Proposed Method

The equipment setup suitable for the proposed method consists of a synchronized and calibrated pair of two identical cameras with the same focal length.

In the proposed method, we exploit a stereo camera pair already calibrated with known calibration features (calibration error). The method relies on existing algorithms of license plate detection. The performance of the license plate detection algorithm affects only the fact whether the speed measurement is performed at all, but does not affect its precision. Using the license plate co-ordinates in a series of images, we first localize the vehicles passing in front of the stereo pair in the series of frames. Consequently, we triangulate the vehicle position in the series of stereo images forming a trajectory using the information known about the stereo setup and the calibration information. Finally, once the trajectory and its individual points are known, we compute the speed (and also acceleration along the trajectory). An overview of the proposed method is shown in Fig. 1.

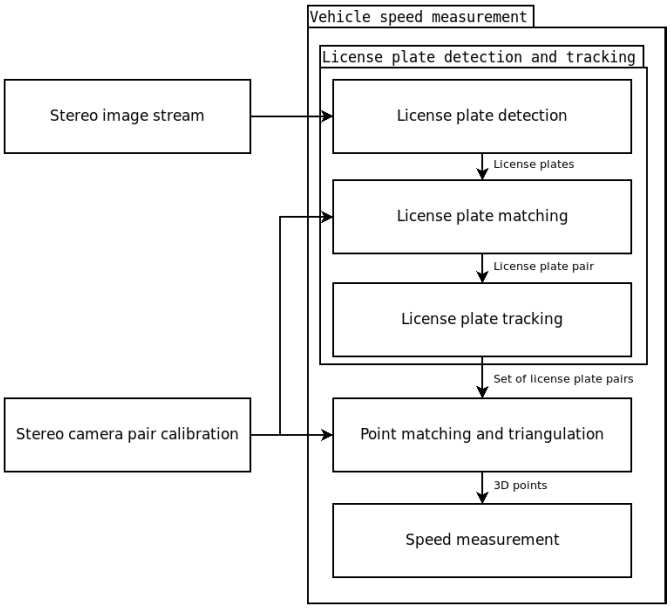



Fig. 1. Overview of the proposed method. Vehicle trajectory is represented using a set of license plate pairs that are extracted from input stereo images. Several points are triangulated along the trajectory using known calibration parameters. Model of vehicle motion is fitted to the triangulated points in order to measure the vehicle speed.

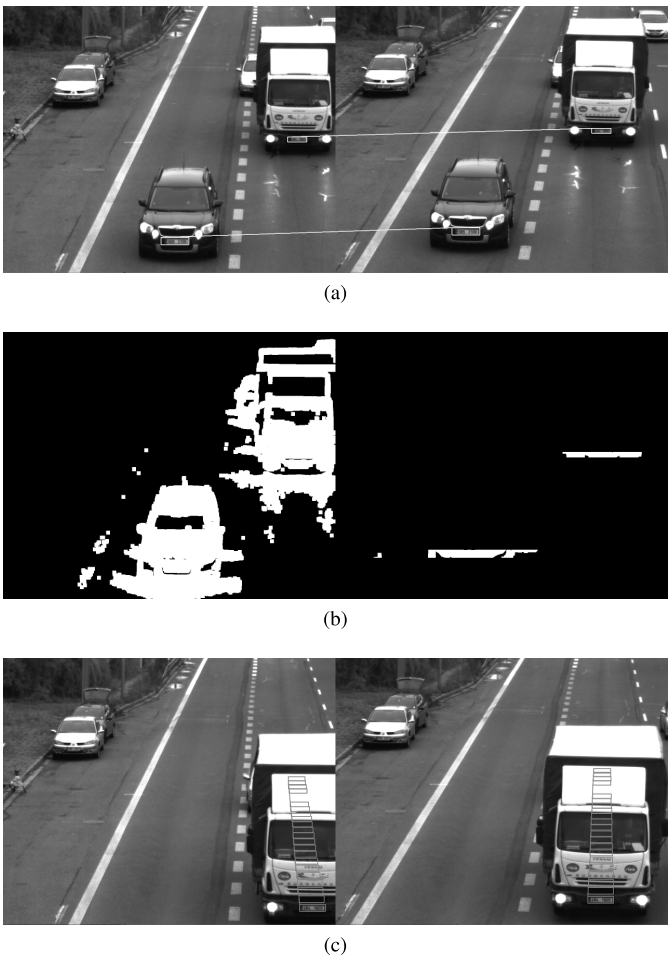
### A. License Plates Detection and Tracking

We assume that each vehicle has a license plate that is firmly attached to its body at a clearly visible place. This assumption allows us to reduce the task of vehicles detection to the task of the license plates detection, which we consider to be much easier due to the standardized appearance of the license plates of a given country. Although the appearances of the license plates differ among countries, they are usually similar enough, so that a detector trained on the license plates of one country is able to detect the license plates from other countries. In the first step of our method, we detect and track the license plates of moving vehicles.

We utilize existing WaldBoost [21] detector with LBP features that was trained to detect license plates with a size of approximately $90 \times 24$ pixels and without or with very small rotation and perspective distortion. Our detector works best for cameras placed on a gate or a bridge above the road looking directly against or with the direction of traffic. Alternatively, the cameras can be placed on a pole on the side of the road, but the angle between the view direction and traffic flow should be kept small. The output of the detector is the top left coordinate and the size of a rectangle that contains the found license plate. License plates in the left and right images are detected separately.

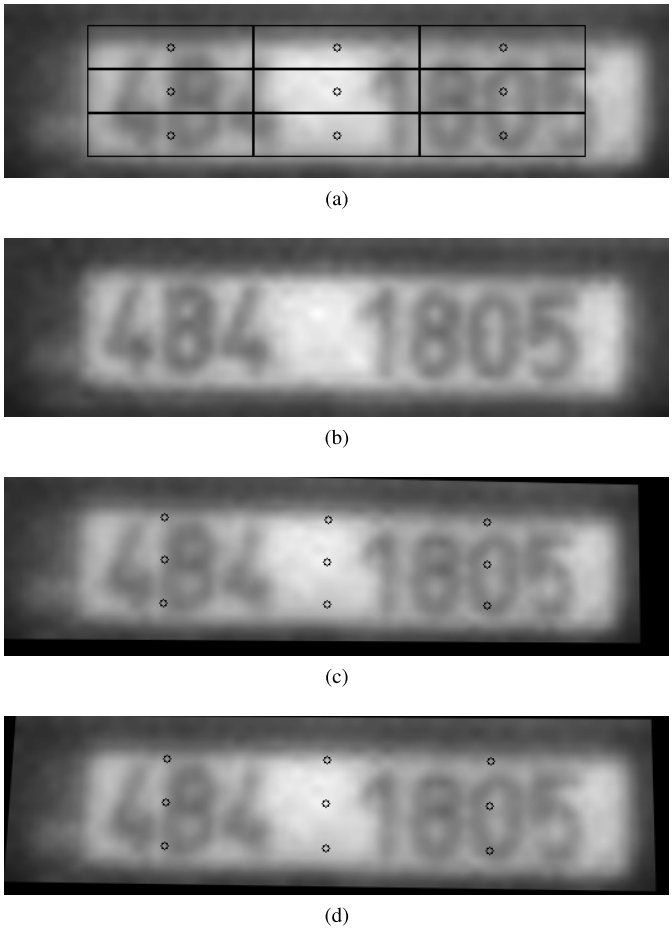
We detect license plates in both images and assign each detected license plate in the left image to a corresponding license plate in the right image, if possible. We assert that the matching license plates are approximately the same size, located at approximately the same position in both images and meet the epipolar constraint. The corresponding license plates then form the license plate pair. For the depiction of detected license plate pairs, see Fig. 2a.

For each license plate pair, we need to decide whether it belongs to a new vehicle or to a vehicle that was seen before. This can be done in numerous ways, for example, using license plate correlation, OCR, optical flow or Kalman filtering. We opted for the latter, that is, license plate tracking using the Kalman filter based on the constant acceleration model. The constant acceleration model approximately corresponds to the motion of the license plates in the images. We chose this approach because it provides a good trade-off between speed, accuracy, and complexity. The approximate predictions supplied by the Kalman filter are enough to perform the license plate re-identification based on its detected and predicted positions in the image and enough to maintain the tracking context in frames where the detection has failed. The tracking in our case is implemented using re-detection and re-identification of license plates in each frame. We maintain a Kalman filter for each vehicle that is currently passing in front of the cameras. Each license plate pair position is checked against all current Kalman filters' predictions. If its position is close enough to some predicted position we assign the checked license plate pair to a set of license plate pairs of the tracked vehicle and update the filter accordingly, otherwise, new vehicle tracking is initialized. When the vehicle passes out of view, we stop



(a) (b) (c)

Fig. 2. License plates detection and tracking. (a) Detected license plates in the image from the camera on the left and their matching counterparts in the right camera image forming a license plate pairs. (b) The license plates search areas in both images were constrained by foreground masks constructed using background subtraction. The right image search area was further constrained using epipolar geometry. (c) A vehicle with its set of all license plate pairs.



(a) (b) (c) (d)

Fig. 3. Point matching and triangulation. (a) Template left image with uniformly sampled points and regions of interests constructed around them. (b) Reference right image. (c) Template image with sampled points warped using rough homography computed from sampled points and their counterparts obtained by template matching the regions of interest to the reference image. (d) Template image with sampled points warped using fine homography computed using enhanced correlation coefficient maximization initialized with rough homography.

the tracking and start processing its set of license plate pairs. A vehicle with its set of license plate pairs is shown in Fig. 2c.

### *B. Point Matching and Triangulation*

After vehicle localization, we are going to triangulate its passage throughout the scene. First, from its set of license plate pairs, we select the pair which contains the largest license plate images. Then, we take the left license plate image of this pair and uniformly sample nine points on it. Around the sampled points we construct small rectangular regions of interest (see Fig. 3a).

Next, we take another license plate pair from the set and match the regions of interest to its left license plate image in order to obtain the points that match the sampled points. To achieve sub-pixel accurate matches, the regions of interest, as well as the template license plate image, are scaled and smoothed prior to template matching. For regions of interest, we use a scale factor of ten. Since the template license plate image is smaller than the sampled license plate image the scale factor used to scale the template image should be greater. Because we do not know the exact value of this scale factor, we try several scales within a reasonable range and choose the one with the highest similarity score. We use the normalized cross-correlation as a similarity metric.

The matched points and the sampled points are used to compute homography transformation between the two left license plate images. This homography transformation is further refined using enhanced correlation coefficient maximization [22]. Finally, the sampled points are transformed using the fine homography in order to obtain fine point matches. These fine point matches represent the sub-pixel accurate left image positions of sampled points in a different time. We repeat the matching process for the rest of the left license plate images from the set and essentially track the sampled points along the vehicle trajectory in left camera images with sub-pixel precision.

So far we have several points in left camera images and in order to perform triangulation, we need to identify their correspondences in right camera images. We can employ a similar

procedure as in the case of finding correspondences among left camera images. First, we take a license plate pair from a set of vehicle license plate pairs and select the left license plate image. Then, we uniformly sample nine points in it, construct the regions of interest, match them to the right license plate image template, compute rough homography and refine it as we did before. Finally, we compute the fine point matches using sampled points from the left license plate image and fine homography transformation. The reference image and template images that are warped using the rough and fine homographies, described above, are shown in Fig. 3b, 3c, and 3d. We repeat this process for the rest of the license plate pairs from the set and receive nine stereo point correspondences for each license plate pair.

Given stereo point correspondences and known internal and external stereo camera pair parameters, we can triangulate the 3D positions of points using existing algorithms. We use the well known Linear-LS method [23].

### C. Speed Computation

To compute the average speed of a passing vehicle, we utilize the triangulated positions from the previous step and their timestamps. First, we correct the triangulated positions by projecting them onto a common plane obtained as a least-square fit through the triangulated points with outliers removed using RANSAC. The corrected points are then subdivided into nine sets in such a way that the same license plate points with different timestamps belong to the same set. These sets are processed separately. The points from a single set, together with their timestamps, serve as an input to a model describing the vehicle motion throughout the scene. We assume that the vehicle is moving with constant or zero acceleration. This type of motion can be described by the following equation:

$$p_i = p + v * \Delta t_i + \frac{1}{2} * a * \Delta t_i^2 \tag{1}$$

where $p_i$ is a co-ordinate of the current position of a corrected triangulated license plate point in time $i$; $p$ is a co-ordinate of initial license plate point position; $v$ is a vector of initial speed; $a$ is a vector of acceleration; and $\Delta t_i$ is the time difference between the current and initial positions. We insert our positional ($p_i$) and time ($\Delta t_i$) data to the model and construct a system of $N * 3$ linear equations, where $N$ is a number of triangulated positions available. This system is usually overdetermined ($N > 3$) and it can be formulated as:

$$A * x = b \tag{2}$$

where $A$ is a $N * 3$ matrix

$$A = \begin{bmatrix} 1 & \Delta t_0 & \frac{1}{2} * \Delta t_0^2 \\ 1 & \Delta t_1 & \frac{1}{2} * \Delta t_1^2 \\ \cdots & & \\ 1 & \Delta t_{N-1} & \frac{1}{2} * \Delta t_{N-1}^2 \end{bmatrix}; \tag{3}$$

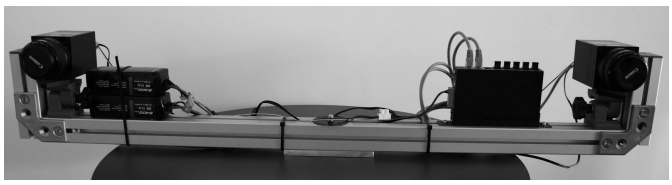


Fig. 4. Our stereo camera pair setup. Two cameras, ethernet switch, and two batteries mounted on an aluminium profile.

$b$ is a $N * 3$ matrix of triangulated positions

$$b = \begin{bmatrix} x_0 & y_0 & z_0 \\ x_1 & y_1 & z_1 \\ \cdots & & \\ x_{N-1} & y_{N-1} & z_{N-1} \end{bmatrix}; \tag{4}$$

and $x$ is a $3 * 3$ matrix of unknown vectors $p$, $v$ and $a$

$$x = \begin{bmatrix} p_x & p_y & p_z \\ v_x & v_y & v_z \\ a_x & a_y & a_z \end{bmatrix}. \tag{5}$$

This system can be solved using various methods for solving linear least square systems such as SVD decomposition. As a result, we obtain an initial point position, initial speed, and acceleration from which we can compute the average speed on the recorded track:

$$v_{avg} = \frac{||A_{N-1} * x - p||}{\Delta t_{N-1}} \tag{6}$$

where $A_{N-1}$ is the last row of matrix $A$. In order to make the computation of the average speed more robust to errors in position triangulation, we employ RANSAC [24] based approach to remove triangulated positions outliers. We repeat this average speed computation process for each of the nine sets with triangulated license plate points and, in the end, we receive nine average speeds. Finally, we select the median of the computed average speeds as the vehicle average speed.

## IV. Evaluation

The properties of our method are evaluated on a dataset we recorded using prototype hardware. We focus the evaluation on the precision and accuracy of speed measurement and then we discuss the cases where the measurement cannot be performed.

### A. Hardware

Our setup (see Fig. 4) consists of two custom made cameras mounted parallelly on a 1 m long aluminium profile placed on a sturdy tripod. The cameras are fitted with PYTHON 1300 global shutter CMOS image sensors and 35mm fixed focal length lens, which is positioned is a such a way that its principal axis is perpendicular to the sensor plane and intersects it at a sensor centre. The image sensors have 0.0048 mm x 0.0048 mm square pixels and provide monochrome $1280 \times 1024$ px images. Raw image data is streamed at a rate of 20 frames per second through a gigabit ethernet switch to a computer where the images are JPEG compressed and stored for further processing. The shutters of the cameras are synchronized using an external trigger with one camera

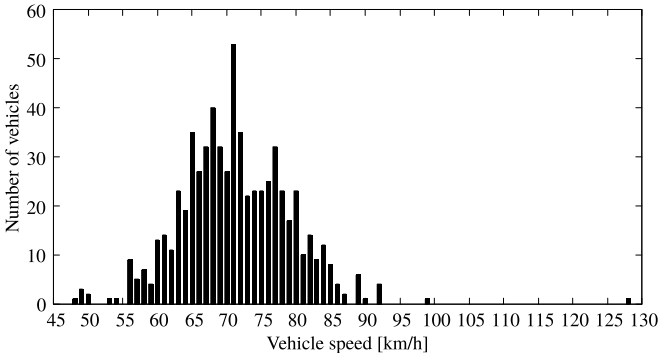



Fig. 5. Histogram of reference average speeds measured by two LIDARs.

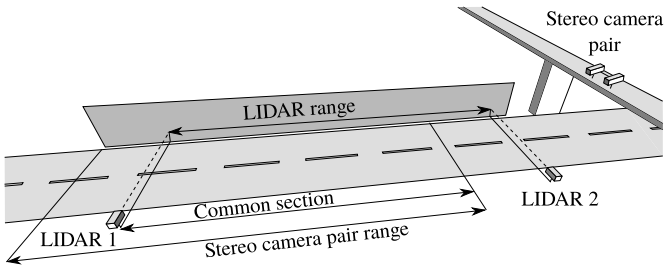



Fig. 6. Schematic drawing of the relative positions of the sensors, their ranges, and their common area.

being the master who sends the trigger signal to the second camera. The cameras and the switch are supplied power from two 6400mAh LiPo batteries attached to the profile.

### B. Dataset

For the purpose of evaluation, we have recorded a dataset using the above-mentioned hardware. The whole dataset was recorded during a single session lasting approximately 40 minutes. During this session, we recorded 654 vehicle passes in two lanes. The left lane (from the point of view of cameras) is fully visible on both cameras while the right lane is only partially visible. The camera setup was placed on a footbridge across the road looking from above towards the incoming vehicles.

To obtain the reference data, we employed the same approach as Sochor *et al.* [25]. We used two LIDARs (LaserAce®IM HR 300) placed at the same height parallel to each other and perpendicularly to the street. The distance $D$ between the LIDARs was 28.05 metres and they were synchronized by the GPS time (Leadtek LR9540D). The distance and time data from both LIDARs was logged in and processed separately. From the logged data, we calculated for each vehicle its immediate speed when entering the first and second laser, its average speed on the distance $D$, its average acceleration, and its length. For more details about measurement process, reference data calculation, and the discussion of measurement error see Sochor *et al.* [25]. The histogram of average speeds of the recorded vehicles over the distance $D$ is shown in Fig. 5.

We want to compare the average speeds measured by the proposed method with the reference average speeds obtained from LIDARs, but those values are comparable if and only if they were both measured over the same section of the road. As the section of the road covered by the two LIDARs and the section of the road in the view of our stereo camera pair do not fully overlap, the reference and the measured average speeds are not directly comparable and we need to adjust them so that the road section, where the speed is measured, is common for both setups. The common section starts at the point where the vehicle enters the first LIDAR and ends at the point where the last vehicle license plate is recorded by both cameras (see Fig. 6). Because our cameras and LIDARs are time-synchronized, we can use their timestamps as a common ground for such an adjustment.

The adjusted reference average speed over the common section of the road is computed as:

$$v_r = v + \frac{1}{2} * a * (t_1 - t_0) \tag{7}$$

where $v$ is the reference immediate speed when entering the first laser; $a$ is the reference acceleration; $t_0$ is the reference time when the vehicle entered the first laser; and $t_1$ is the time when the last license plate of the vehicle was recorded.

The adjusted measured average speed over the common section of the road is computed as:

$$\begin{aligned} p_0 &= p + v * (t_0 - t) + \frac{1}{2} * a * (t_0 - t)^2 \\ p_1 &= p + v * (t_1 - t) + \frac{1}{2} * a * (t_1 - t)^2 \\ v_m &= \frac{||p_1 - p_0||}{t_1 - t_0} \end{aligned} \tag{8}$$

where $p_0$ and $p_1$ are co-ordinates of vehicle positions at times $t_0$ and $t_1$; $t_0$ is the reference time when the vehicle entered the first laser; $t_1$ is the time when the last license plate of the vehicle was recorded; $p$ is a co-ordinate of the initial vehicle position at time $t$; $v$ is the co-ordinate of immediate vehicle speed at time $t$; and $a$ is the co-ordinate of vehicle acceleration. The values of $p$, $v$, and $a$ are obtained from a vehicle motion model (Eq. 5).

### C. Timestamp Assignment Latency

One more thing should be considered, and that is the delay between the end of camera exposure and the timestamp assignment, which, in our case, takes place in the computer that stores the frames. The cameras we used, as soon as the exposure ends, pack the read-out lines into the UDP packets and send them to the computer where they are received by the software. The timestamp is assigned immediately after receiving the first UDP packet of a new frame, and it is the same for both images.

We measure the timestamp assignment latency by pointing the cameras on a series of LEDs which encode the millisecond part of current time in a binary format (see Fig. 7). We examine the recorded frames and compare the time encoded in LEDs to the millisecond portion of the frame timestamp. In our

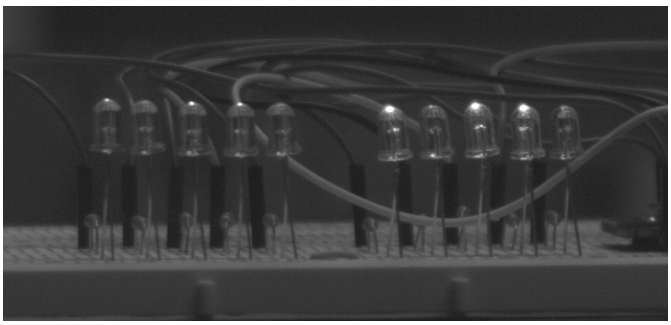


Fig. 7. A camera image cutout that contains blinking LEDs. This image was assigned a timestamp with millisecond portion of 729. The LEDs in the image encode millisecond time just after the end of exposure which was 726 (whose binary image is 10110 10110 as seen above). The latency of timestamp assignment, in this case, is therefore 3 ms.

case, the timestamp assignment latency has a mean value of 2.5 ms and a standard deviation of 1 ms. The measured values are used to correct the recorded frame timestamps in Eq. 8.

### D. Implementation

We broke the proposed method pipeline into five smaller steps and implemented them as separate programs. These programs correspond to License plate detection and matching, License plate tracking, Point matching, Triangulation, and Speed measurement. The pipeline was implemented in C++ with the help of OpenCV and Boost libraries. The programs pass the data along the pipeline in JSON format through standard streams.

During the implementation, we focused more on the precision and accuracy of the measurement rather than on the optimality of the implementation or the speed of computation. We performed only a few optimizations that were aimed mostly at reducing the detection time of licence plates by limiting the image area in which the license plates were going to be detected. To limit the license plate detection area, we first performed the background subtraction for each image of the stereo image pair. The foreground image masks we obtained are then morphologically dilatated to close the holes. After that, the license plates are detected in the masked left image. For each license plate found in the left image, we compute an epipolar line in the right image using a detected rectangle top left corner and fundamental matrix. We use the computed epipolar lines to further limit the search area in the right image. The foreground masks are shown in Fig. 2b. Finally, we detect the license plates in the masked right image.

In any case, we evaluate the average time of execution for each implemented step separately. The evaluation takes place on a Linux desktop computer with an Intel Core i5-6500 processor running at 3.2 GHz with 24 GB RAM. The data is obtained using a sample vehicle recording that consists of 20 frames in which its license plate is successfully detected. The results are summarized in Table I. The total processing time currently exceeds the 50 ms time limit for real-time processing of 20 frames per second supplied by the cameras. The most time demanding step, despite our simple optimization, is still the License plate detection and matching step and it has become the prime candidate for further optimization or implementation in hardware.

TABLE I
AVERAGE PER FRAME EXECUTION TIME OF INDIVIDUAL PIPELINE STEPS

| Step | Time [ms] |
|---|---|
| License plate detection and matching | 219.5 |
| License plate tracking | 3.5 |
| Point matching | 196.5 |
| Triangulation | 4.5 |
| Speed measurement | 9.8 |

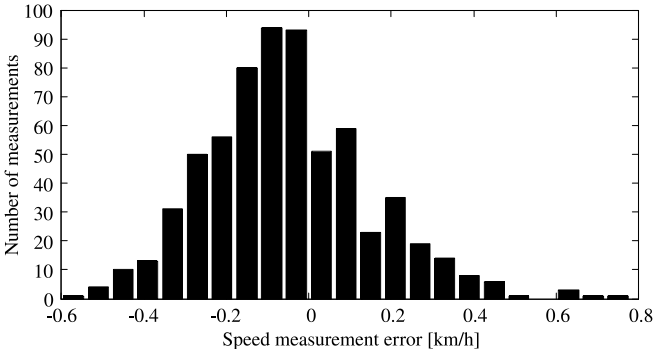



Fig. 8. Histogram of speed measurement errors.

The total processing time per stereo frame with a single passing vehicle is approximately 0.45 seconds. The average number of stereo frames that we process per vehicle in our dataset is 13. Therefore, we can process a single vehicle in less than 6 seconds. If the average time distance between the two incoming vehicles is more than our processing time, we would be able to process all of them in near real-time. The mean time distance between the two incoming vehicles in our dataset acquired during "peak times" is approximately 3 seconds. Processing the traffic of this volume in near real-time would require cutting the processing time per vehicle to half or, for example, adding another computing unit. However, the traffic volume changes during the day, and, given enough storage, we can utilize the low volume periods to catch up with the processing of the stored frames. Because the real-time processing is usually not a requirement for traffic enforcement systems, we should be able to compute the speed for all the passing vehicles as long as the traffic volume on a given location is less than approximately 14400 vehicles per day (600 per hour).

### E. Speed Measurement

We evaluate the accuracy and precision of speed measurement using the above-mentioned dataset. The speed is measured for 653 of a total of 654 recorded vehicles. One measurement is missing because the license plate detector failed to detect the vehicle license plate. Detailed results for all vehicles in the dataset can be accessed online.[1] From the measured values we compute the measurement error as:

$$e = v_m - v_d \tag{9}$$

[1] http://www.stud.fit.vutbr.cz/ xnajma00/results.json

TABLE II
COMPARISON OF STEREO AND TWO-CAMERA BASED VEHICLE SPEED MEASUREMENT METHODS

| | MSE [km/h] | Max abs. error [km/h] | Max perc. error [%] | Dataset size | Number of different vehicles |
|---|---|---|---|---|---|
| Llorca et al. [14] | NA | 2.62 | NA | 64 | 1 |
| Jalalat et al. [9] | NA | NA | 3.3 | 441 | 441 |
| El Bouziady et al. [11] | 2.33 | 2.00 | NA | 12 | 6 |
| Yang et al. [12] | 0.42 | -1.60 | 3.80 | 56 | 2 |
| **Proposed method** | **0.04** | **0.72** | **1.11** | **653** | **653** |

where $v_m$ is average speed measured using the proposed method and $v_d$ is reference average speed from our dataset. For the histogram of the speed measurement errors see Fig. 8. Overall, the measured speed has a maximum negative error of -0.56 km/h and a maximum positive error of 0.72 km/h. The mean error is -0.05 km/h with a standard deviation of 0.20 km/h. The mean absolute percentage error is 0.23 % and maximum percentage error is 1.11 %.

We compare the speed measurement errors with three other stereo-based vehicle speed measurement methods and one two-camera method mentioned in chapter II, namely, Jalalat *et al.*'s method [9], El Bouziady et al's method [11], Yang *et al.*'s method [12], and Llorca *et al.*'s [14] method. The comparison is shown in Table II. The proposed method achieves better results than the other methods in all compared statistics on a much bigger and more diverse dataset.

## V. CONCLUSION

We proposed a novel method for vehicle speed measurement using a stereo camera pair. First, the method detects vehicle license plates in both images and matches them together. Detected license plate pairs are tracked in the following stereo images and the vehicle trajectory is reconstructed. After that, several points are uniformly sampled on the left license plate image of the largest recorded license plate pair on vehicle trajectory. The same points are then found in the remaining left and right license plate images using template matching. Next, the left and right corresponding points are triangulated and their co-ordinates are used in the mathematical model that describes vehicle motion. Finally, the system of equations is constructed, and upon solving, the vehicle speed is obtained.

Using our prototype hardware that consists of two synchronized cameras, ethernet switch, and two batteries mounted on one meter long aluminium profile, we recorded a dataset. The dataset contains recordings of 654 vehicles for which the reference average speeds were provided by a pair of LIDARs. We use the reference average speeds and the average speeds measured by the proposed method to compute the measurement errors. The results were compared to the existing stereo-based and two-camera based methods.

The newly proposed method measures the speed of the passing vehicles more precisely than the other methods. One metric is especially important - the standard deviation - only in our case it is less than 1 km/h that is required to meet the $\pm 3$ km/h tolerance with 99.9 % probability. Not exceeding the standard deviation of 1 km/h during metrological testing is necessary for the device to become acknowledged by the metrological legislation of EU as a measurement device. The standard deviation and the maximum absolute error in our case are 0.2 km/h and 0.72 km/h, respectively. These values suggest that our method could be suitable for devices that should comply with even stricter metrological legislations than those enforced in the EU.

Future research should focus on optimization and acceleration of individual pipeline steps. In our implementation, the total execution time of all pipeline steps exceeds 50 ms which makes this implementation unsuitable for real time processing of 20 frames per second. As the license plate detection and point matching are the two most time consuming steps, we suggest experimenting with different types of detectors and matching algorithms and/or exploiting hardware acceleration.

Apart from enhancing the presented pipeline, one may also focus on maintaining the correct camera calibration in time. This calibration correction may be necessary for any long-term installed speed measurement stereo-based device as the initial calibration is likely to change due to the external influences such as temperature changes or tremors.

Our dataset was recorded in the morning hours during good weather conditions. Therefore future work could also focus on expanding the limited scope of situations present in the dataset. The situations that can be included are severe weather conditions, including the presence of rain, snow, or dust and otherwise varying weather conditions.

## REFERENCES

[1] O. Publishing and I. T. Forum, *Towards Zero: Ambitious Road Safety Targets and the Safe System Approach*. Paris, France: Organisation for Economic Co-operation and Development, 2008.

[2] K. Bhalla *et al.*, "Rapid assessment of road safety policy change: Relaxation of the national speed enforcement law in russia leads to large increases in the prevalence of speeding," *Injury Prevention*, vol. 21, no. 1, pp. 53–56, Feb. 2015, doi: 10.1136/injuryprev-2014-041189.

[3] L. A. Klein *et al.*, "Traffic detector handbook: Volume I," Turner-Fairbank Highway Research Center, McLean, VA, USA, Tech. Rep. FHWA-HRT-06-108, 2006.

[4] P. Beyer, "Non-intrusive detection, the way forward," in *Proc. Southern African Transp. Conf.*, 2015, pp. 879–888.

[5] P. T. Martin *et al.*, "Detector technology evaluation," Mountain-Plains Consortium, Fargo, ND, USA, MPC Rep. 03-154, 2003.

[6] M. P. Muresan, S. Nedevschi, and R. Danescu, "A multi patch warping approach for improved stereo block matching," in *Proc. 12th Int. Joint Conf. Comput. Vis., Imag. Comput. Graph. Theory Appl.*, 2017, pp. 459–466.

[7] R. Spangenberg, T. Langner, S. Adfeldt, and R. Rojas, "Large scale semi-global matching on the CPU," in *Proc. IEEE Intell. Vehicles Symp. Proc.*, Jun. 2014, pp. 195–201.

[8] V. Kolmogorov and R. Zabih, "Graph cut algorithms for binocular stereo with occlusions," in *Handbook of Mathematical Models in Computer Vision*. Boston, MA, USA: Springer, 2006, pp. 423–437.

[9] M. Jalalat, M. Nejati, and A. Majidi, "Vehicle detection and speed estimation using cascade classifier and sub-pixel stereo matching," in *Proc. 2nd Int. Conf. Signal Process. Intell. Syst. (ICSPIS)*, Dec. 2016, pp. 1–5.

[10] Z. Zhang, "A flexible new technique for camera calibration," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 22, no. 11, pp. 1330–1334, Nov. 2000.
[11] A. El Bouziady, R. O. H. Thami, M. Ghogho, O. Bourja, and S. El Fkihi, "Vehicle speed estimation using extracted SURF features from stereo images," in *Proc. Int. Conf. Intell. Syst. Comput. Vis. (ISCV)*, Apr. 2018, pp. 1–6.
[12] L. Yang, M. Li, X. Song, Z. Xiong, C. Hou, and B. Qu, "Vehicle speed measurement based on binocular stereovision system," *IEEE Access*, vol. 7, pp. 106628–106641, 2019.
[13] H. Bay, A. Ess, T. Tuytelaars, and L. Van Gool, "Speeded-up robust features (SURF)," *Comput. Vis. Image Understand.*, vol. 110, no. 3, pp. 346–359, Jun. 2008. [Online]. Available: https://linkinghub.elsevier.com/retrieve/pii/S1077314207001555
[14] D. F. Llorca *et al.*, "Two-camera based accurate vehicle speed measurement using average speed at a fixed point," in *Proc. IEEE 19th Int. Conf. Intell. Transp. Syst. (ITSC)*, Nov. 2016, pp. 2533–2538. [Online]. Available: http://ieeexplore.ieee.org/document/7795963/
[15] M. Dubska, A. Herout, and J. Sochor, "Automatic camera calibration for traffic understanding," in *Proc. Brit. Mach. Vis. Conf.*, vol. 4, no. 6, 2014, p. 8.
[16] J. Sochor, R. Juránek, and A. Herout, "Traffic surveillance camera calibration by 3D model bounding box alignment for accurate vehicle speed measurement," *Comput. Vis. Image Understand.*, vol. 161, pp. 87–98, Aug. 2017. [Online]. Available: https://linkinghub.elsevier.com/retrieve/pii/S1077314217301108
[17] J. Sochor, J. Spanhel, and A. Herout, "BoxCars: Improving fine-grained recognition of vehicles using 3-D bounding boxes in traffic surveillance," *IEEE Trans. Intell. Transp. Syst.*, vol. 20, no. 1, pp. 97–108, Jan. 2019. [Online]. Available: https://ieeexplore.ieee.org/document/8307405/
[18] D. C. Luvizon, B. T. Nassu, and R. Minetto, "A video-based system for vehicle speed measurement in urban roadways," *IEEE Trans. Intell. Transp. Syst.*, vol. 18, no. 6, pp. 1393–1404, Jun. 2017.
[19] M. Famouri, Z. Azimifar, and A. Wong, "A novel motion plane-based approach to vehicle speed estimation," *IEEE Trans. Intell. Transp. Syst.*, vol. 20, no. 4, pp. 1237–1246, Apr. 2019. [Online]. Available: https://ieeexplore.ieee.org/document/8401851/
[20] A. Bartoli, Y. Gérard, F. Chadebecq, T. Collins, and D. Pizarro, "Shape-from-template," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 37, no. 10, pp. 2099–2118, Oct. 2015. [Online]. Available: http://ieeexplore.ieee.org/document/7010934/
[21] J. Sochman and J. Matas, "WaldBoost—Learning for time constrained sequential detection," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2005, pp. 150–156.
[22] G. D. Evangelidis and E. Z. Psarakis, "Parametric image alignment using enhanced correlation coefficient maximization," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 30, no. 10, pp. 1858–1865, Oct. 2008. [Online]. Available: http://ieeexplore.ieee.org/document/4515873/
[23] R. I. Hartley and P. Sturm, "Triangulation," *Comput. Vis. Image Understand*, vol. 68, no. 2, pp. 146–157, 1997.
[24] M. A. Fischler and R. C. Bolles, "Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography," in *Readings in Computer Vision*, M. A. Fischler and O. Firschein, Eds. San Francisco, CA, USA: Morgan Kaufmann, 1987, pp. 726–740. [Online]. Available: http://www.sciencedirect.com/science/article/pii/B9780080515816500702
[25] J. Sochor *et al.*, "Comprehensive data set for automatic single camera visual speed measurement," *IEEE Trans. Intell. Transp. Syst.*, vol. 20, no. 5, pp. 1633–1643, May 2019. [Online]. Available: https://ieeexplore.ieee.org/document/8356199/

**Pavel Najman** received the M.S. degree from the Brno University of Technology (BUT), Brno, Czech Republic, where he is currently pursuing the Ph.D. degree with the Department of Computer Graphics and Multimedia, Faculty of Information Technology. His research interests include computer vision, particularly stereo camera pair calibration, and scene reconstruction in the context of traffic surveillance.

**Pavel Zemčík** (Member, IEEE) received the M.Sc. and Ph.D. degrees from the Brno University of Technology (BUT), Brno, Czech Republic. He is currently a Professor and the Dean of the Faculty of Information Technology (BUT), Brno. His research interests include image and signal processing, computer graphics, computer vision, implementation of the image related algorithms in embedded systems, and applications of such algorithms in industry as well as traffic enforcement and surveillance.