CrossMark

Development of a tracking-based system for automated traffic data collection for roundabouts

Hai Dinh[1] · Hua Tang[2]

Abstract Traffic data collection is essential for performance assessment, safety improvement and road planning. While automated traffic data collection for highways is relatively mature, that for roundabouts is more challenging due to more complex traffic scenes, data specifications and vehicle behavior. In this paper, the authors propose an automated traffic data collection system dedicated to roundabout scenes. The proposed system has mainly four steps of processing. First, camera calibration is performed for roundabout traffic scenes with a novel circle-based calibration algorithm. Second, the system uses enhanced Mixture of Gaussian algorithm with shaking removal for video segmentation, which can tolerate repeated camera displacements and background movements. Then, Kalman filtering, Kernel-based tracking and overlap-based optimization are employed to track vehicles while they are occluded and to derive the complete vehicle trajectories. The resulting vehicle trajectory of each individual vehicle gives the position, size, shape and speed of the vehicle at each time moment. Finally, a data mining algorithm is used to automatically extract the interested traffic data from the vehicle trajectories. The overall traffic data collection system has been implemented in software and runs on regular PC. The total processing time for a 3-hour video is currently 6 h. The automated traffic data collection system can significantly reduce cost and improve efficiency compared to manual data collection. The extracted traffic data have been compared to accurate manual measurements for 29 videos recorded on 29 different days, and an accuracy of more than 90% has been achieved.

Keywords Traffic data collection · Vehicle tracking · Roundabout · Vision-based systems · Intelligent transport systems

1 Introduction

Traffic data collection is very important in transportation applications to assess performance, improve safety and design roads [1]. Before modern powerful computing systems are economically available, traffic engineers or human operators were traditionally deployed in the field for manual traffic data collection. For example, hand-held intersection counter [2] can be used to collect turning traffic volumes at an intersection. However, clearly this manual process is very challenging, time-consuming and costly. For example, when traffic is relatively heavy at an intersection, a traffic engineer may not be able to simultaneously count volume for all turning directions. In the past two decades, image sensors associated with the ever-increasing power of modern computing systems are increasingly affordable, and hence there has been widespread deployment of camera-based vision systems for traffic monitoring, traffic management, traffic data collection, traffic accident warning, etc. [3–6]. Among many versatile applications of these camera-based vision systems, one of them is automated traffic data collection, which can significantly improve efficiency and reduce cost compared to manual data collection.

In the literature, there has been a significant amount of work on automated traffic data collection by processing the

✉ Hua Tang
htang@d.umn.edu

1 GeoCue Group, Inc., 9668 Madison Blvd #202, Madison, AL 35758, USA

2 University of Minnesota Duluth, 271 MWAH, 1023 University Dr., Duluth, MN 55812, USA

recorded videos from camera-based vision systems. Those systems/tools (with underlying algorithms and methodologies) developed for highways or arterial roads are relatively successful in terms of accuracy of automated traffic data collection [7–10]. One important contributing factor to the good accuracy of these systems is the relatively simple vehicle behavior in highways or arterial roads and relatively simple traffic data to be collected. For example, a sample picture of highway traffic from [7] is shown in Fig. 1 and in typical scenarios, highway vehicles move at relatively constant speed in one straight direction with relatively in-frequent acceleration/de-acceleration behavior and they do not turn, yield or stop unless there is severe congestion. Besides, associated with relatively simple vehicle behavior, interested traffic data to be collected for highways are usually vehicle speed, vehicle volume, lane use/change and vehicle classification [7–10].

In contrast to highways or arterial roads, vehicles at signalized/un-signalized intersections or roundabouts have very different behavior and interested traffic data are also very different and more complex to collect. The main traffic scene of interest in this work is roundabouts. A roundabout is a type of circular intersection or junction in which vehicles always move in one circular direction and it requires entering vehicles to give way to vehicles already inside the circle. A sample picture of a roundabout is given in Fig. 1 as well. It has been studied and shown that roundabouts have many benefits compared to signalized intersections such as improved traffic flow and safety [11, 12]. However, traffic data collection for roundabouts is more challenging compared to highways, due to fundamentally physical differences in traffic scenes [13, 14]. In contrast to usually straight parallel traffic lanes in highways as shown in Fig. 1, the circular shape of the roundabout inherently causes more complex vehicle behavior. Besides,

roundabouts have always quite a few entrances and exits, which significantly complicates the vehicle behavior due to the need to yield (while highways have entrances and exits as well, vehicle behavior is typically much simpler). Compared to highways, vehicles entering or inside the roundabout are more likely to encounter acceleration/de-acceleration, stopping, waiting and turning. These behaviors may present significant challenges for accurate and reliable vehicle tracking in a camera-based vision system to derive vehicle trajectories. The more complex vehicle behavior also translates to more complex traffic data to be collected. Interested traffic data for roundabouts include not only speed, volume, vehicle classification as in highways, but also origin–destination pairs, waiting time and gap size that are sort of unique to the roundabouts.

With more roundabouts being designed especially in suburban or rural areas, traffic data collection for these traffic scenes is in great need to assess capacity, performance and safety [14]. Of particular, importance among all types of interested traffic data is the gap size, which is defined as the minimum headway in the circulating traffic that is accepted by a driver desiring to enter the roundabout [15]. Gap size may be further refined to accepted gap size or rejected gap size. Figure 2 shows a picture of the same roundabout in Fig. 1 with focus on two of the main entrances/exits when the camera was panned with a different angle. Figure 2 illustrates one case of accepted gap size as the vehicle A enters the roundabout (by passing line 2) while it needs to yield to the vehicle B. Clearly, gap size is an important performance measure of a roundabout and a smaller gap size would mean that the roundabout performs better by carrying more traffic [15]. Compared to signalized intersections, drivers desiring to enter the roundabout have to make own subjective decisions on whether it is safe to enter instead of relying on external timing signals.

Fig. 1 Picture of a sample highway traffic from Ref. [7] (**a**) and a sample roundabout (**b**)
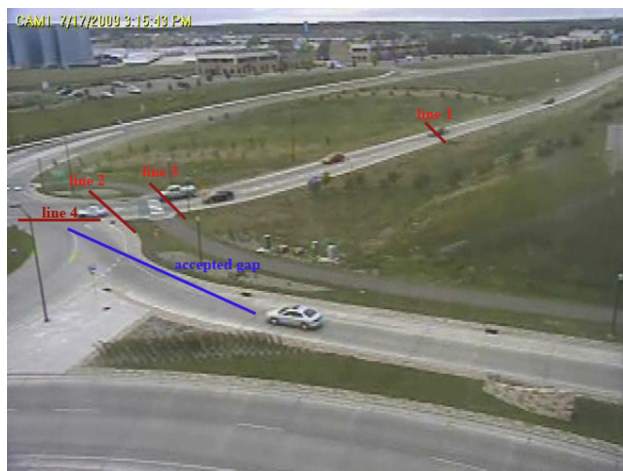
Fig. 2 Illustration of computation of the gap size

Therefore, gap size is also a very important measure of safety in a roundabout [15–18]. As so, gap size is one of the most wanted type of traffic data for roundabouts.

Among interested traffic data to be collected for the roundabouts, it may be possible to manually collect vehicle volume in the field by the deployment of human traffic engineers. However, as in highway case, manual collection is very time-consuming as mentioned before. In fact, on one hand such a hand-held counter as the one used in intersections [2] is not available yet so the traffic engineer may have to manually record on paper, and on the other hand it is very challenging to be able to track multiple vehicles at the same time, because all entrances are open and vehicles from all entrances may be entering the roundabout at the same time (unlike signalized intersections where usually only two turning directions are open at the same time). Regarding gap size, it is very difficult and very error-prone to collect them if manual collection is at all possible [14]. For example, referring to Fig. 2, a traffic engineer must first record when the vehicle A enters the roundabout and then start to count how long it takes for the vehicle B to reach line 2, in order to collect just one sample of accepted gap size. It becomes much harder to collect rejected gaps. Therefore, to collect a large database of gap sizes for statistical measurement purpose, the manual approach is not realistic. One alternative approach is to use camera-based vision system to pre-record a video of the roundabout traffic at time of interest, such as peak hours, and then traffic engineers manual inspect the video to collect accepted/rejected gap size [14]. While this approach is viable and saves some effort compared to manual data collection in the field, it is still extremely time-consuming and costly. In our experience, it takes on average at least a day to manual inspect and collect a few hundred samples of gap sizes for a 3-hour video (the exact time needed depends on the number of samples of accepted/rejected gap sizes in the video).

As manual traffic data collection is deemed infeasible as discussed above, the need for automated traffic data collection for roundabouts arises. Unlike relatively developed systems/tools for automated data collection for highways, those for automated traffic data collection for roundabouts are relatively scarce in the literature. The most relevant work to automated traffic data collection for roundabouts is those for signalized/un-signalized intersections [5, 6, 19, 20]. In [5, 6], wireless sensors were designed and used for detection of individual vehicle passing similar to how the loop inductor works for detection and counting of highway traffic. A sensor is typically placed close to the stopping line of each lane (but in the middle of the lane), and a vehicle was detected and recorded when it drove past it. When a vehicle makes a turn, ideally it is first detected by the sensor from the source lane and then by the sensor from the destination lane. In this way, vehicle turning volumes, which are a very desirable type of traffic data for intersections, can be automatically collected. However, the sensor approach has a fundamental algorithm limitation in that a cross-turning (for instance from North to East) may not be distinguished from a right turn (for instance from South to East), which results in inherent counting errors [19]. Besides, the sensor approach may collect only turning volumes, not other types of traffic data such as vehicle speed, waiting time and accepted/rejected gap sizes. If extending the sensor approach for intersections to roundabouts, the situation would be similar that only vehicle volume from entrances/exits may be collected. Therefore, the senor approach for intersections is not considered acceptable for traffic data collection of roundabouts.

Excluding the sensor approach, a very viable approach is to adopt the camera-based vision systems for automated traffic data collection for roundabouts like those for highways [7–10] or those for intersections [5, 6]. It may appear very straightforward to apply those developed vision systems/tools for highways or intersections to roundabouts. While indeed the concept is the same that camera-based vision systems can be applied to record videos which are then video processed to automatically extract interested traffic data for roundabouts, the systems must be dramatically modified or extended in order to accommodate the specifics of roundabouts, such as more complex vehicle behavior and very different traffic data to be collected than highways as discussed before. For example, in [5] the traffic data collection system for intersections is limited to vehicle volume, speed and waiting time, while accepted/rejected gap sizes have not been tackled. Therefore, motivated by the need to automatically collect gap size and other traffic data for roundabouts, in this work the authors

🍎 Springer

J. Mod. Transport. (2017) 25(1):12–23

develop a system/tool dedicated to automated traffic data collection for roundabouts. To the best of our knowledge, this work may be one of the early efforts to allow automated accepted/rejected gap size collection for roundabouts.

The rest of the paper is organized as follows. In Sect. 2, an overview of the proposed system for automated data collection for roundabouts is provided, and in Sect. 3 we describe the detailed processing steps. In Sect. 4, we present experimental results on a real-world roundabout and finally conclusions are drawn in Sect. 5.

2 Overview of the proposed data collection system

The proposed system for automated traffic data collection for roundabouts takes recorded videos of roundabout traffic as inputs. It is assumed that a camera-based vision system is installed nearby the roundabout to record videos. In our work, the videos are pre-recorded and stored electronically as.avi or.xvid files, then supplied into the traffic data collection system as inputs. In other words, the developed system is currently processing videos offline. However, online processing of the videos is very approachable if the computing systems, such as a regular PC, were integrated into the camera-based vision systems, which is beyond the scope of this paper.

The proposed traffic data collection system has mainly two functional modules, namely the tracking module and the data mining module. Once the pre-recorded videos are supplied in, the tracking module is responsible for
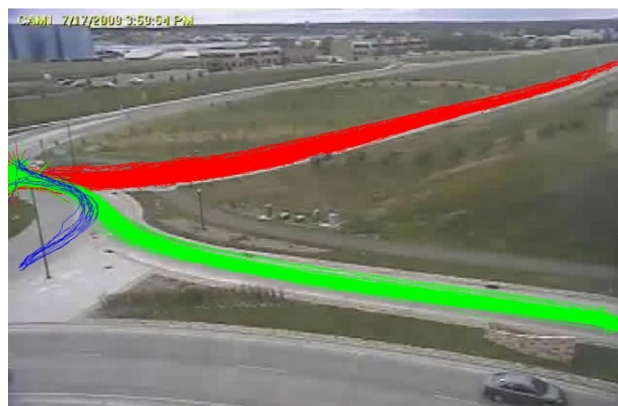

Fig. 4 Overlay of vehicle trajectories (one line represents one vehicle trajectory)

processing the video to derive the raw data of vehicle trajectories, and then the data mining module mines the vehicle trajectories to extract interested traffic data. For each module, the processing is mostly automated with minimal requirements of manual setting or inputs from the user. Note that we derive the raw data of vehicle trajectories as they provide most comprehensive traffic information. A vehicle trajectory gives the position of the vehicle at each time moment indexed by the image frame, and the positions allow to estimate vehicle speed and acceleration/de-acceleration behavior and also allows to determine whether the vehicle has entered the roundabout. Further analysis of the vehicle trajectory would also allow to derive waiting time and accepted/rejected gaps, which will be detailed later in Sect. 3.
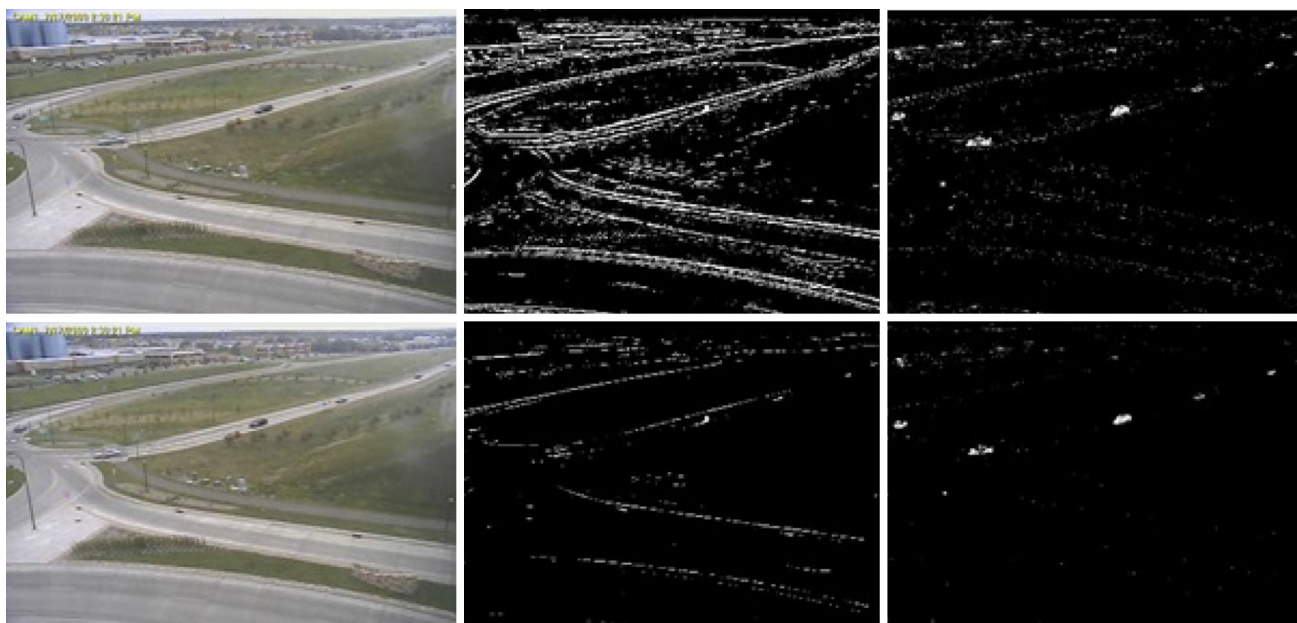

Fig. 3 Segmentation results with MoG and shaking-removal in case of camera shaking
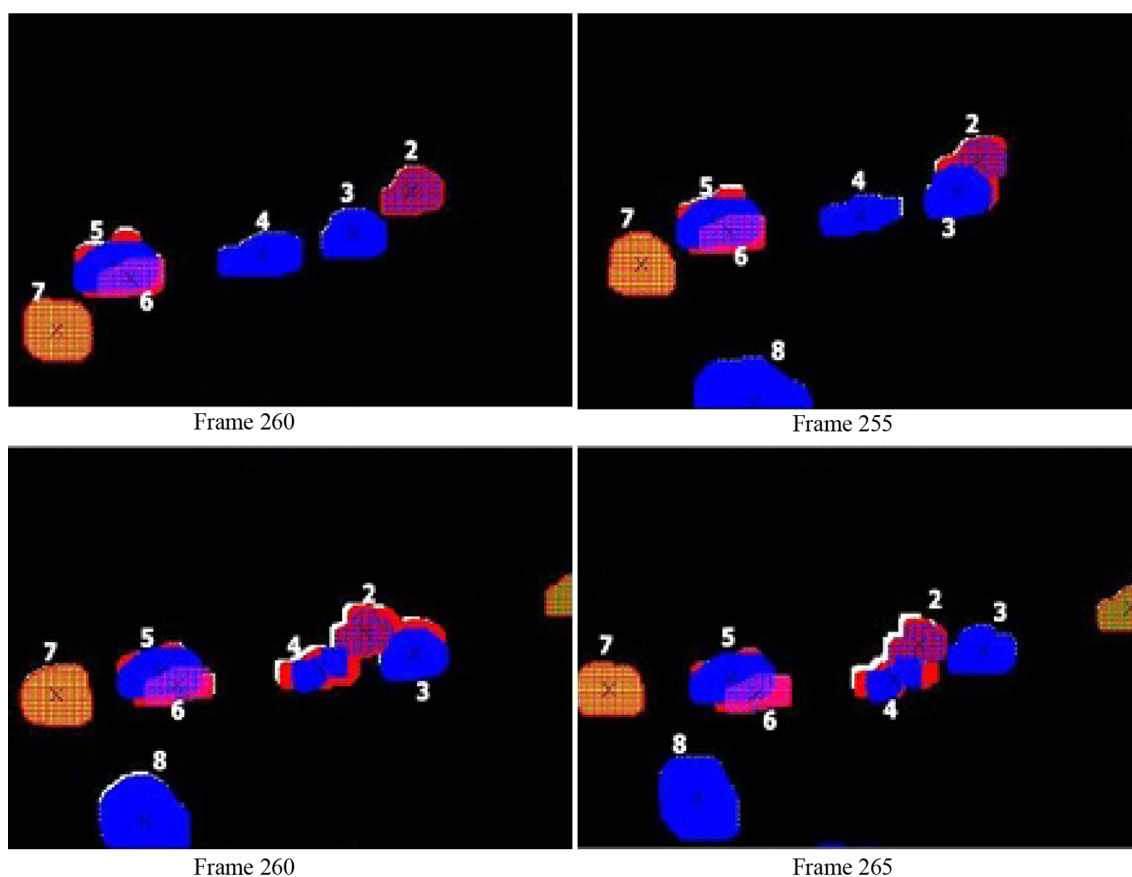
Fig. 5 Tracking with vehicle occlusions

To drive the vehicle trajectories, there are three major processing steps in the tracking module that incorporate powerful image/video processing techniques/algorithms. The first processing step is camera calibration (as is always the case with any camera-based vision system) that allows to establish the relation between image dimension (or distance) in pixels to real-world dimension (or distance) in meters/feet. Clearly, this relation is required to estimate real-world vehicle speed and vehicle length/width. After camera calibration, the next major processing step in the tracking module is vehicle segmentation, which is to detect/identify the vehicles from image frames. There are many established algorithms for vehicle segmentation and the Mixture of Gaussian (MoG) algorithm was adopted in our work together with a proposed camera shaking-removal algorithm. Once segmented vehicles are obtained, the last major processing step is vehicle tracking, which is to associate or link detected vehicles across all image frames. Like vehicle segmentation, there are quite some vehicle tracking algorithms reported in the literature and the proposed one in our work is a combined region-based tracking with kernel-based tracking algorithm. We will detail the above three processing steps in Sect. 3.

The outputs from the tracking module are raw data of vehicle trajectories. Each vehicle trajectory contains the position information of that vehicle across image frames. The data mining module is then invoked to process all vehicle trajectories to extract interested traffic data, including speed, volume, waiting time and accepted/rejected gaps. While speed and volume data may seem trivial to extract from simple manipulation of the position data per individual trajectory, waiting time and especially accepted/rejected gaps require further analysis of the position data across multiple vehicle trajectories, which will be detailed in Sect. 3 as well.

For the proposed system for automated traffic data collection for roundabouts to be practically useful, accuracy of the collected traffic data is of most priority while processing time (or in general demands of computing power or resources) is considered secondary. In our work, the traffic data from the proposed system were compared to ground truth measurements to evaluate the accuracy. It should be noted that between the two modules, the tracking module has significant impact on the accuracy of traffic data than the data mining module. In fact, the data mining module can faithfully extract the traffic data given inputs of vehicle
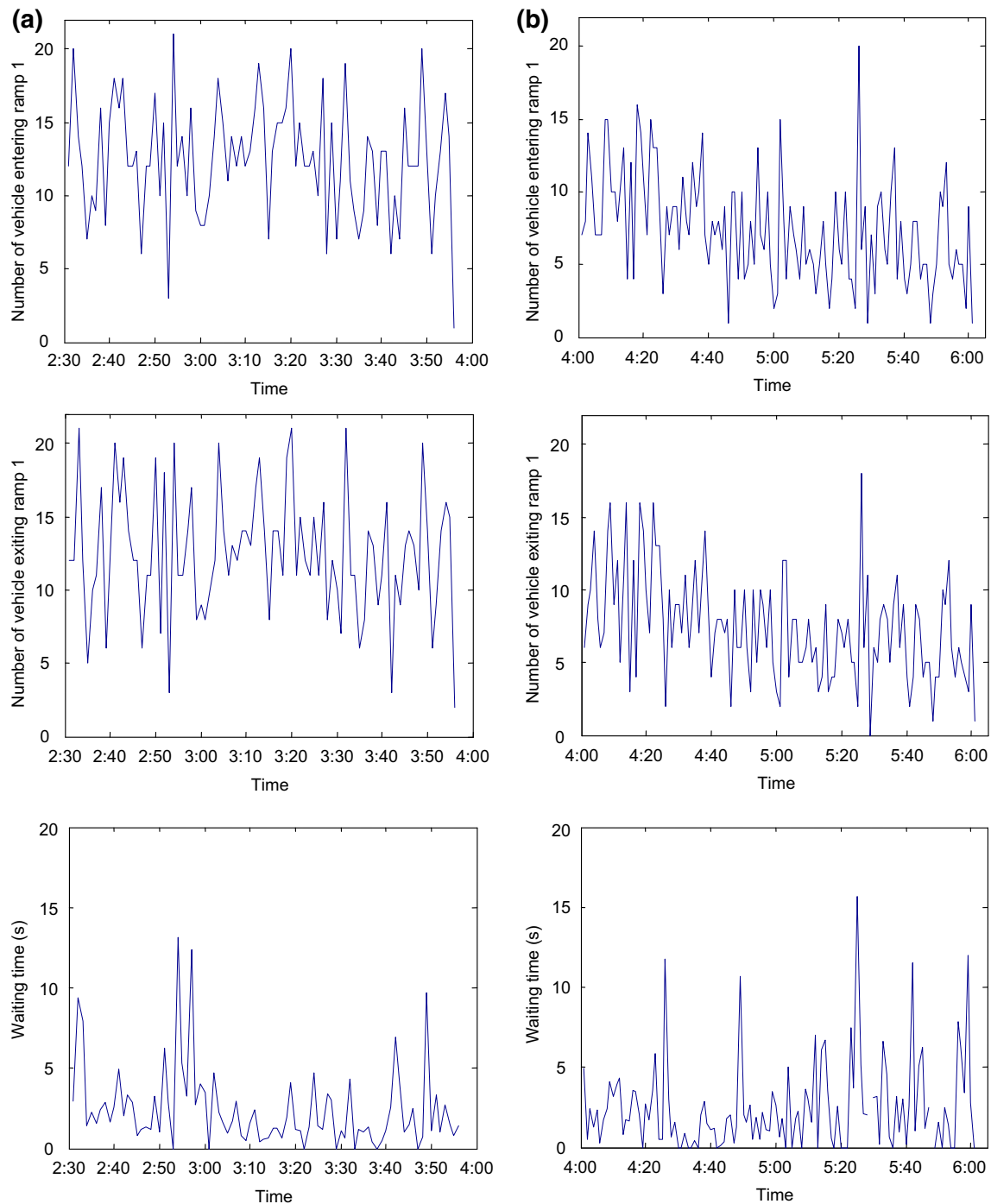
Fig. 6 Number of vehicles entering/exiting the ramp and waiting time in each minute for two videos collected on 17th 2009. **a** During 2:30–3:55 pm. **b** During 4:00–6:00 pm

trajectories and the main target for data mining is efficiency (or the processing time). However, the tracking module derives the vehicle trajectories, whose accuracy will fundamentally determine the accuracy of collected traffic data. It is well known that camera calibration, vehicle segmentation and vehicle tracking all introduce error when using a camera-based vision system, especially the later two steps [5, 6].

3 The proposed system for automated traffic data collection

As discussed in Sect. 2, the proposed system for automated traffic data collection for roundabouts consists of the tracking module and the data mining module. Each module is presented in detail below.
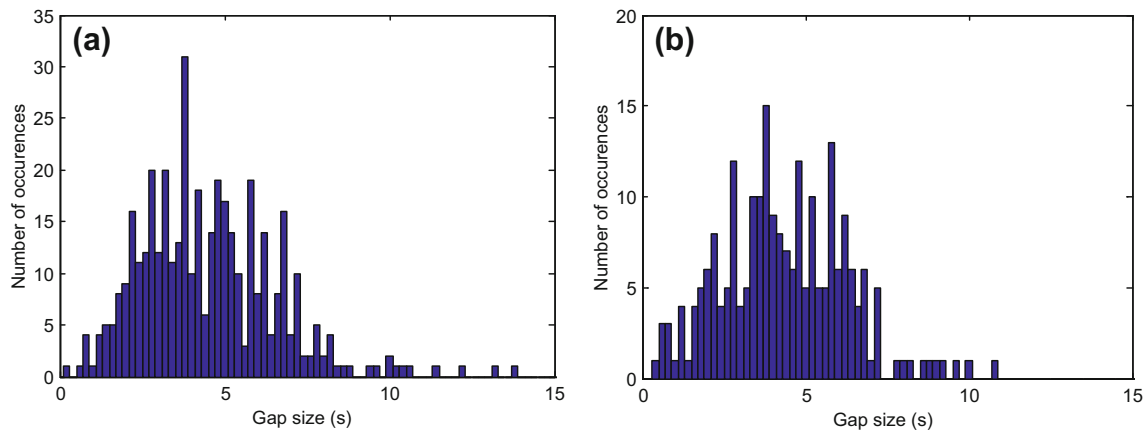
Fig. 7 Histogram of accepted gap sizes for two videos collected on July 17th 2009. **a** During 2:30–3:55 pm. **b** During 4:00–6:00 pm

3.1 The tracking module

The tracking modules have mainly three processing steps: camera calibration, vehicle segmentation and vehicle tracking, which will be discussed below sequentially.

3.1.1 Camera calibration

While camera calibration methods are well studied for highways [21–23], they do not apply to roundabouts due to different features available in the scene. Previous work on camera calibration of highways mostly takes advantage of parallel traffic lanes [21–23]. However, in general, parallel traffic lanes are not available at roundabouts and instead circular lanes exist (for instance the circles in Fig. 1). Therefore, we proposed to use the available landmark features of circles for camera calibration of roundabouts in [24]. The idea is briefly discussed below.

Using a camera geometry setup in [24], the image coordinates (i_x, i_y) is projected to the world coordinates (w_x, w_y) by the following equations

$$w_x = -\frac{hi_x}{f\sin\varphi + i_y\cos\varphi}, \tag{1}$$

$$w_y = \frac{hi_y}{\sin\varphi(f\sin\varphi + i_y\cos\varphi)}, \tag{2}$$

where h is the camera height, f the focal length and φ the tilt angle. The equation for a circle in the real-world coordinate is

$$(w_x - a)^2 + (w_y - b)^2 = R^2, \tag{3}$$

where (a, b) denotes the center of the circle and R the radius, which is usually available from geometric design of the roundabout (for instance, $R = 50$ feet for the roundabout in Fig. 1). On the other hand, the equation for an ellipse in the image coordinate is

$$i_x^2 + 2Hi_xi_y + Bi_y^2 + 2Gi_x + 2Fi_y + C = 0, \tag{4}$$

where $H, B, G, F,$ and C are coefficients. With perspective transformation characterized by Eqs. (1) and (2), a real-world circle characterized by Eq. (3) becomes an ellipse in the image characterized by Eq. (4) [11]. Therefore, substituting Eqs. (1) and (2) into Eq. (3), the resulting equation should match exactly Eq. (4). By coefficient matching of the resulting equation from (3) against (4), camera parameters h, f and φ can be solved. Note that Eq. (4) can be derived from ellipse-fitting a number of manually selected (or automatically detected) pixels in the image that belong to the ellipse [24].

It is worth noting that the above method does not require a complete visible circle and a partial circle works too. For example, in Fig. 2, one can see that only a partial landmark of a circle is visible in the image and this allows camera calibration using the proposed method in [24]. However, one should note that a complete visible circle (if available) may have better accuracy for camera calibration [24].

3.1.2 Vehicle segmentation

Vehicle segmentation is one of the most important steps of any video-based data collection system in that it is responsible for detecting/identifying vehicles and its accuracy has a significant impact on the vehicle tracking accuracy and eventually the overall traffic data collection accuracy. In our work, we adopted the MoG algorithm for vehicle segmentation that was originally proposed in [25], among many possible other options [26–32].

The MoG algorithm considers the values of a pixel at a particular position (i_x, i_y) of an image over time t as a pixel process, and the recent history of the pixel is modeled by a mixture of K Gaussian distributions. The probability of observing a value of X_t is [25, 33]:

$$P(X_t) = \sum_{i=1}^{K} w_{i,t} \times \eta\big(X_{i,t}, \mu_{i,t}, \Sigma_{i,t}\big), \qquad (5)$$

where X_t stands for the incoming pixel at time t (or image frame t), $w_{i,t}$ the weights factor, η a Gaussian probability density function, $\mu_{i,t}$ the mean value and $\Sigma_{i,t}$ the covariance matrix of the ith Gaussian distribution at time t ($\Sigma_{i,t} = \delta_{i,t}^2 \boldsymbol{I}$, where $\delta_{i,t}^2$ denotes the variance of the ith Gaussian distribution at time t and $\boldsymbol{I}$ the identity matrix). The sum of weights of the K Gaussian distributions at any time t is normalized to 1.0. At any time t, the portion of the Gaussian distributions (B out of K) that accounts for the background is defined to be

$$B = \operatorname{argmin}_b \left(\sum_{i=1}^{b} w_{i,t} T \right), \qquad (6)$$

where T, the threshold, is a measure of the minimum portion of the data that is used to account for the background. The rest of Gaussian distributions is used for foreground model.

At each image frame t, for each pixel, the new observation of the pixel is matched against each Gaussian distribution of that pixel. A match to the ith Gaussian distribution is defined as the new observation X_t of the pixel within the interval of J times the standard deviation off the mean, i.e.,

$$p_{i,t} = \big|X_t - \mu_{i,t}\big|/\delta_{i,t} \le J, \qquad (7)$$

If the new observation X_t of the pixel does not match any of the K Gaussian distributions, X_t will be declared as a foreground for the current pixel, and the MoG model is updated by simply replacing the mean of distribution with the lowest weight by X_t and initializing the variance with a typical value (for instance 25 pixel square) while keeping the same weight. The other $(K-1)$ distributions would have their mean, variance and weight kept the same.

If on the other hand the new observation of X_t of the pixel matches at least one of the K Gaussian distributions, the best-matched distribution (i.e., the one with the minimum $p_{i,t}$), is used for foreground/background declaration. If the best-matched distribution, say r, belongs to the portion of Gaussian distributions that account for background, then X_t is declared background. Otherwise, it is declared foreground. In either case, the best-matched distribution is updated by increasing its weight and "learning" its mean and variance for the next image frame $t+1$ as follows [25]:

$$\left. \begin{aligned} w_{r,t+1} &= (1-\alpha)w_{r,t} + \alpha \\ \mu_{r,t+1} &= (1-\beta)\mu_{r,t} + \beta X_t \\ \delta_{r,t+1}^2 &= (1-\beta)\delta_{r,t}^2 + \beta\big(X_t - \mu_{r,t}\big)^2 \end{aligned} \right\}, \qquad (8)$$

where α and β are the learning rates to update weight, mean and variance. For the rest of Gaussian distributions, they will have their weights decreased and mean/variance kept the same, as follows [25] (where $i \ne r$):

$$\left. \begin{aligned} w_{i,t+1} &= (1-\alpha)w_{i,t} \\ \mu_{i,t+1} &= \mu_{i,t} \\ \delta_{i,t+1}^2 &= \delta_{i,t}^2 \end{aligned} \right\}. \qquad (9)$$

After update of each Gaussian distribution, the value of B in Eq. (5) is re-calculated as well. Finally, the K distributions are sorted in weights for matching operations in the next image frame $t + 1$.

We adopted the MoG algorithm for video segmentation for a few considerations. First, compared to other alternatives, the MoG algorithm may achieve a better tradeoff between demands of computing power/resources and segmentation accuracy, as discussed in [34]. Second, repeated camera shaking (for instance due to constant wind) is a very practical issue in camera-based vision systems (as the authors had experienced in recorded videos) in that it causes noisy segmentation (to be illustrated in Sect. 4) and can affect the accuracy of the following step of vehicle tracking. The MoG handles repeated camera shaking very well as repeated observation changes of a pixel due to camera shaking are very likely to be modeled in background due to its inherent multi-modal modeling capability [25]. However, in case of sudden camera shaking (for instance due to gust), MoG per se may not help as it has not had enough observations to build up the multi-modal background distributions. In that case, the authors employ a shaking-removal step that was proposed in [35]. The idea is to compare the observations of pixels of the detection region against the background distributions of both current detection region and a small neighborhood region (of the detection region). If the pixel of the detection region matches with the background distributions of the neighborhood region, it is highly likely that the current observation of the pixel is a background from the neighborhood region as opposed to a foreground in the current detection region. In our work, we empirically chose a square 5×5 window for the neighborhood region among choices of 4×4, 5×5, and 6×6, which experimentally all gave similar results. In general, from our experience working with various videos, we recommend choose a window size that is between 1/100 and 1/25 of the segmented vehicle size in the image.

Table 1 Average accuracy in vehicle count, waiting time and gap size in comparison to manual measurements

	(%)		
	Vehicle count	Waiting time	Gap size
Average accuracy	95.50	92.00	100.00

3.1.3 Vehicle tracking

After vehicle segmentation, the next step is to track vehicle to derive the complete vehicle trajectory, which are the desired raw data that are later used to extract interested traffic data. In this sub-section, a description of how to track vehicles from the outputs of vehicle segmentation is given.

Among some reported methods for vehicle tracking [36–38], we propose to combine region-based tracking [8] with kernel-based tracking [39, 40]. After vehicle segmentation, results are binary blobs in the image and these blobs are extracted and classified as vehicles if they meet at least the threshold size. A state vector is associated with each valid vehicle and it records the position of vehicle at each image frame. In addition to the positions, other information of the vehicle, such as size and vehicle shape/contour, can be recorded and be used to classify vehicles if needed. Given the current state of a vehicle at image frame t, we use Kalman filtering to predict the state of a vehicle in the next image frame $t + 1$ [41]. To associate vehicles between frame t and frame $t + 1$, the algorithm compares the segmented vehicles in frame $t + 1$ (i.e., the target) against the vehicles from frame t (i.e., the model) in joint feature-spatial spaces using the Kernel-based tracking algorithm [39, 40]. The feature-spatial model of a vehicle is characterized in image frame t and predicted for comparison against the target in frame $t + 1$. Finally, note that the state vector, which contains the vehicle position at each image frame, gives the complete trajectory of a vehicle once a vehicle is tracked.

Compared to the traditional region-based tracking [8] alone, the combined algorithm gives more accuracy in vehicle tracking at the expense of computational time, especially in the case of vehicle occlusion thanks to the joint feature-spatial model of a vehicle that provides more evidence for vehicle association in addition to the regions.

3.1.4 Data mining

The results from three previous processing steps are raw data of vehicle trajectories, from which a comprehensive data mining algorithm can then be used to extract interested traffic data, such as vehicle speed, volume, waiting time and accepted/rejected gap size. From these trajectories, vehicle volume and speed could be readily computed.

As for waiting time, it can be derived by subtracting free-flow time from travel time, while travel time again is easily obtained by counting how many image frames it takes for a vehicle from entering the ramp to entering the roundabout.

As automated collection of gap size has not been reported before in the literature, we detail on this type of data collection. To facilitate the collection of accepted and rejected gaps, we first manually drew a few lines from road markers as shown in Fig. 2. We consider that a vehicle A from the ramp entrance entered the roundabout when it crossed line 2. If this happened while there were other vehicles B in the other entrance (which has right-of-way) or inside the roundabout itself, we would collect one sample of accepted gap size, which is the travel time from when vehicle A crossed line 2 to when the other vehicle B crossed line 4. Similarly, to collect rejected gaps, we consider that vehicles A from the ramp entrance waited to enter the roundabout when they crossed line 3 but not line 2 yet. If there were vehicles B in the other entrance or inside the roundabout while vehicle A were in the waiting mode, we would collect one sample of rejected gap size, which is the travel time from when vehicle A crossed line 3 to when the vehicles B crossed line 4. If there are multiple vehicles B involved in accepted/rejected gap, the one with the shortest travel time is taken.

This proposed approach to compute accepted/rejected gap size involves manual setup of a few lines for computation purpose, which is a slightly bit of extra work, however, this approach is very reliable and gives very accurate data. Future works will look into how to remove manual setup of the lines.

4 Experimental testing results

In this section, we present experiment results from practical testing of the proposed system for automated traffic data collection for roundabouts. The system has been implemented in C and Matlab and runs on a regular PC. We tested the system using 29 videos of a roundabout recorded on 29 different days. The roundabout is located in Cottage Grove, Washington County of Minnesota USA as shown in Fig. 2, and the video was recorded from typical surveillance cameras installed by Minnesota Department of Transportation [42]. The videos were recorded at 7 frames per second with a resolution of 640×480, and the average video length was 3 h. The total processing time for a 3-hour video was on average 6 h. To make the proposed system run in real-time, image resolution could be lowered to 352×288 or 320×240 as used by many traditional systems. Another option is to simplify the MoG algorithm for video segmentation and especially the complex vehicle tracking algorithm.

First, the camera calibration results are discussed briefly. The measured camera height (51 feet) is very close to the calibrated result (52.3 feet) with 2.5% error. With the calibrated camera parameters, vehicle speeds were computed and the results agreed well with the measured speeds with an error less than 10%.

Next, results from vehicle segmentation are shown below. As shown in Fig. 3, the left column shows two consecutive image frames that have encountered significant camera shaking, and the middle column shows segmentation results with low and high thresholds in traditional background subtraction methods [26], and finally the right column segmentation results from the proposed method. It can be recognized the combined MoG algorithm with camera shaking-removal are very effective to reduce noisy segmentation regions, which would help improve tracking accuracy.

In most cases, vehicles are well tracked and the complete trajectory of a vehicle is obtained. Figure 4 shows the overlay of all vehicle trajectories derived from a 4-hour video. A valid or a correct vehicle trajectory in our work is considered to be one that corresponds to a real-world vehicle in the video. The correct vehicle trajectories are 93% of the total tracked ones. The main factors that affect this accuracy are occasionally very large camera shaking, significant light changes, and long and significant vehicle occlusions at times. Light to moderate vehicle occlusions are handled well by the combined region-based and kernel-based tracking algorithm used in our work. Figure 5 gives an example of tracking under occlusions. Vehicle 2 merges with vehicle 3 first and they together merges with vehicle 4, but each was individually tracked under occlusions. Also notice vehicles 5 and 6 had significant occlusions (while they were waiting to enter the roundabout), due to an existing vehicle 7 inside the roundabout and vehicle 8 from the other ramp entrance (which actually gave a case of rejected gap).

Next, the results from the data mining module that further processes the raw data of tracked vehicle trajectories are shown below. Figure 6 shows the vehicle volume that entered and exited the ramp 1 (in red in Fig. 4) in every minute for two videos. The average waiting time to enter the roundabout from ramp 1 in every minute was shown in Fig. 6 as well. One can clearly notice the longer waiting time at about 5:30 pm, which corresponded to the rush hour. Another longer waiting time was observed at about 2:55 pm, when it was found that roundabout had more vehicles inside. Figure 7 shows the histogram of accepted gap sizes for the same videos. Clearly, accepted gap size peaks at about 4 to 5 s, which is the headway that most drivers are comfortable with when deciding to enter the roundabout. This gap size is very consistent to the findings reported in other works [15, 43, 44].

To quantitatively measure the accuracy of the proposed traffic data collection system, we inspected the videos and manually counted and recorded the number of vehicles entering and exiting the ramp, which gives most accurate ground truth data. Then, the collected data from the proposed system were compared against the manually collected data for accuracy estimation. Table 1 summarizes the average accuracy. The vehicle count accuracy was over 95%, and the accuracy on average waiting time was about 92%. The gap size accuracy is almost 100%. As previously mentioned in Sect. 3.1.3, given raw data of vehicle trajectories from the tracking module, the data mining module does not incur any accuracy loss when extracting traffic data. The error of the collected traffic data is strictly from some erroneous vehicle trajectories from the tracking module. The main source of error in the tracking module was occasional poor detection/segmentation and significant vehicle occlusions between vehicles. For example, in Fig. 5, vehicle 5 and 6 had significant occlusions while waiting to enter the roundabout from the ramp. Consider the case that vehicle 5 in fact did not enter the roundabout but was mistakenly tracked so (regardless of what happened to vehicle 6) due to occlusion and poor detection/segmentation (especially after long wait), and this would cause under-estimated waiting time and a false accepted gap size for vehicle 5. On the other hand, if vehicle 5 in fact entered the roundabout but was not tracked so, it would result in over-estimated waiting time or a miss of accepted gap size for vehicle 5. Examples of more difficult cases were that if only vehicle 5 entered the roundabout but the tracking module mistakenly tracked both so, or instead only vehicle 6 so. However, it is worth noting that in spite of possible false gap sizes or misses of gap sizes in the above cases, their effect on the accepted/rejected gap sizes is statistically minimal, as the number of false ones plus misses was less than 5% of the total number of sample gap sizes collected (about 11,000 from 29 videos).

5 Conclusion

In this paper, the authors proposed a system for automated traffic data collection for roundabouts. The developed system consists of a tracking module and a data mining module. The tracking module has three major processing steps of camera calibration, vehicle segmentation and vehicle tracking. Landmark features of circles typically available at roundabouts were used for camera calibration, and MoG algorithm together with shaking-removal were adopted for vehicle segmentation, and finally a combined region-based tracking with kernel-based tracking algorithm was proposed to derive vehicle trajectories. The data mining module then processes the raw data of vehicle

J. Mod. Transport. (2017) 25(1):12–23

 Springer

trajectories to automatically collect traffic data, such as vehicle volume, waiting time and accepted/rejected gaps. Extensive experiments on a real-world roundabout have verified the correct operation of the system, and the accuracy of automated data collection was over 90% accuracy compared to ground truth measurements.

Such a developed system is very valuable to traffic engineers as it allows automated traffic data collection so that they do not have to manually collect traffic data using either hand-held devices in the field or pre-recorded traffic videos, which significantly improves efficiency and reduces cost. It also helps address the increasing need of traffic data collection for roundabouts, especially the gap size.

In future work, the authors plan to target traffic data collection for the complete roundabout. However, this requires a complete and good coverage of the roundabout, which is difficult using a single camera. Especially when the camera is not mounted very high above the ground, significant vehicle occlusions from those ramps that are furthest away from the camera will pose a challenge for accurate vehicle tracking. One viable option is to use multiple cameras for better coverage of the complete roundabout and surrounding ramps. As with any other camera-based vision system, the developed system shares a few common limitations, which are briefly noted below. First, adverse weather conditions (such as heavy rain, snow or fog) may cause noisy detection/segmentation, which further affects the tracking accuracy and eventually traffic data accuracy. In our work, the videos were mostly recorded on overcast days. Second, vehicle shadow may cause noisy detection/segmentation as well and is always a challenge in vehicle tracking.

References

1. St-Aubina P, Sauniera N, Miranda-Morenob L (2015) Large-scale automated proactive road safety analysis using video data. Transp Res Part C Emerg Technol Part B 58:363–379
2. http://diamondtraffic.com/productlist/Portable-Counters
3. Muffert M, Uwe W (2013) A stereo vision based object tracking approach at roundabouts. IEEE Intell Transp Syst Mag 5(2):22–32
4. Romanoni A, Mussoneb L, Rizzia D, Matteuccia M (2015) A comparison of two Monte Carlo algorithms for 3D vehicle trajectory reconstruction in roundabouts. Pattern Recognit Lett 51:79–85
5. Shirazi M, Morris B (2016) Vision-based turning movement monitoring: count, speed & waiting time estimation. IEEE Intell Transp Syst Mag 8(1):23–34
6. Datondji S, Dupuis Y, Subirats P, Vasseur P (2016) A survey of vision-based traffic monitoring of road intersections. IEEE Trans Intell Transp Syst 17(10):2681–2698
7. Unzueta L, Nieto M, Cortes A, Barandiaran J, Otaegui O, Sanchez P (2012) Adaptive mutlicue background subtraction for robust vehicle counting and classification. IEEE Trans Intell Transp Syst 13(2):527–540
8. Smith C, Richards C, Brandt S, Papanikolopoulos NP (1996) Visual tracking for intelligent vehicle highway systems. IEEE Trans Veh Technol 45(4):744–759
9. Dailey D (2006) CCTV Technical Report: Phase III, Washington Department of Transportation. Technical report, WA/RD-2006-635.2
10. Ervin R, MacAdam C, Walker J, Bogard S, Hagan M, Vayda A, Anderson E (2000) System for Assessment of the Vehicle Motion Environment (SAVME). University of Michigan Transportation Research Institute, Final Report Volume II
11. Roundabout Benefits, available at https://www.wsdot.wa.gov/Safety/roundabouts/benefits.htm
12. Case for Modern Roundabouts, available at https://priceonomics.com/the-case-for-more-traffic-roundabouts
13. Miovision Technologies Inc. 2016. www.miovision.com, Ontario, Canada
14. Roundabouts and Data Collection, available at http://miovision.com/wp-content/uploads/Roundabout_Data_Collection_Slides.pdf
15. Mensah S, Eshragh S, Faghri A (2010) A critical gap analyses of modern roundabouts. In: Transportation Research Board Annual Meeting, Washington
16. Fitzpatrick C, Abrams D, Tang Y, Knodler M Jr (2013) Spatial and temporal analysis of driver gap acceptance behavior at modern roundabouts. Transp Res Rec J Transp Res Board 2388(2013):14–20
17. Ahmad S (2014) Capacity-related driver behavior on modern roundabouts built on high-speed roads", Master't thesis, Purdue University
18. Zheng D, Chitturi M, Bill A, Noyee D (2012) Critical gaps and follow-up headways at congested roundabouts, In: Transportation Research Board Annual Meeting, Washington
19. Kwon T (2008) Portable Cellular Wireless Mesh Sensor Network for Vehicle Tracking in an Intersection. Minnesota Department of Transportation, Final report, CTS 08-29
20. Abdel-Rahim A, Johnson B (2008) An intersection traffic data collection device utilizing logging capabilities of traffic controllers and current traffic sensors. National Institute for Advanced Transportation Technology, Final report
21. Kanhere NK, Birchfield ST, Sarasua WA (2008) Automatic camera calibration using pattern detection for vision-based speed sensing. Transp Res Rec J Transp Res Board 2086:30–39
22. Pumrin S, Dailey DJ (2002) Dynamic camera calibration in support of intelligent transportation systems. Transp Res Rec J Transp Res Board 1084:77–84
23. Schoepflin TN, Dailey DJ (2003) Dynamic camera calibration of roadside traffic management cameras for vehicle speed estimation. IEEE Trans Intell Transp Syst 4(2):90–98
24. Dinh H, Tang H (2014) A simple method for camera calibration for roundabout traffic scenes using a single circle. IET Intell Trans Syst 8(3):175–182
25. Stauffer C, Grimson WEL (1999) Adaptive background mixture models for real-time tracking. IEEE Conf Comput Vis Pattern Recognit 2:246–252

26. Gupte S, Masound O, Martin RFK, Papanikolopoulos NP (2002) Detection and classification of vehicles. IEEE Trans Intell Transp Syst 3(1):37–47
27. Wang L, Yung NHC (2010) Extraction of moving objects from their background based on multiple adaptive thresholds and boundary evaluation. IEEE Trans Intell Transp Syst 11(1):40–51
28. Hsieh JW, Yu SH, Chen YS (2006) Automatic traffic surveillance system for vehicle tracking and classification. IEEE Trans Intell Transp Syst 7(2):175–186
29. Lam W, Pang C, Yung N (2004) Highly accurate texture-based vehicle segmentation method. Opt Eng 43(3):591–603
30. Heikkila M, Pietikainen M (2006) A texture-based method for modeling the background and detecting moving objects. IEEE Trans Pattern Anal Mach Intell 28(4):657–662
31. Lai A, Yung N (1998) A fast and accurate scoreboard algorithm for estimating stationary backgrounds in an image sequence. Proc IEEE Intl Symp Circuit Syst 4:241–244
32. Oliver N, Rosario B, Pentland A (2000) A bayesian computer vision system for modeling human interactions. IEEE Trans Pattern Anal Mach Intell 22(8):831–843
33. Pavlidis I, Morellas V, Tsiamyrtzis P, Harp S (2001) Urban surveillance systems: from the laboratory to the commercial world. Proc IEEE 89(10):1478–1496
34. Toyama K, Krumm J, Brumitt B, Meyers B (1999) Wallflower: Principls and Practices of Background Maintenance. In: Proceeding of IEEE International Conference on Computer Vision and Pattern Recognition, pp 255–261
35. Elgammal A, Harwood D, Davis L (2000) Non-parametric Model for Background Subtraction. In: Proceeding of European Conference on Computer Vision, Part II, pp. 751–767
36. Yiu B, Wong K, Chin F, Chung R (2005) Explicit Contour Model for Vehicle Tracking with Automatic Hypothesis Validation. IEEE International Conference on Image Processing
37. Tan TN, Sullivan GD, Baker KD (1998) Model-based localization and recognition of road vehicles. Int J Comput Vis 27(1):5–25
38. Beymer D, McLauchlan P, Coifman B, Malik J (1997) A real-time computer vision system for measuring traffic parameters.In: IEEE Conference on Computer Vision and Pattern Recognition, pp. 495–501
39. Comaniciu D, Ramesh V, Meer P (2003) Kernel-based object tracking. IEEE Trans Pattern Anal Mach Intell 25(5):564–575
40. Yang C, Duraiswami R, Elgammal A, Davis L (2005) Real-time Kernel-Based Tracking in Joint Feature-Spatial Spaces. In: IEEE Conference on Computer Vision and Pattern Recognition, pp. 1–7
41. Welch G, Bishop G (2006) An Introduction to the Kalman filter. http://www.cs.unc.edu/~welch/media/pdf/kalman_intro.pdf
42. Mn-DoT (2009) Minnesota Department of Transportation, www.dot.state.mn.us
43. Mandavilli S (2005) Study of Operational Performance and Environmental Impacts of Modern Roundabouts in Kansas. In: Transportation Research Board National Roundabout Conference
44. Parma K, Denholm J III (2006) Going Around the Neighborhood—a Roundabout Case Study in Texas.In: Proceeding of IET Annual Meeting and Exhibit Compendium of Technical Papers