A Multi-Loop Vehicle-Counting Method under Gray Mode and RGB Mode

Yue Chen and Jian Lu

Abstract: With the rapid development of road traffic, real-time vehicle counting is very important in the construction of intelligent transportation systems (ITSs). Compared with traditional technologies, the video-based method for vehicle counting shows great importance and huge advantages in its low cost, high efficiency, and flexibility. However, many methods find difficulty in balancing the accuracy and complexity of the algorithm. For example, compared with traditional and simple methods, deep learning methods may achieve higher precision, but they also greatly increase the complexity of the algorithm. In addition to that, most of the methods only work under one mode of color, which is a waste of available information. Considering the above, a multi-loop vehicle-counting method under gray mode and RGB mode was proposed in this paper. Under gray and RGB modes, the moving vehicle can be detected more completely; with the help of multiple loops, vehicle counting could better deal with different influencing factors, such as driving behavior, traffic environment, shooting angle, etc. The experimental results show that the proposed method is able to count vehicles with more than 98.5% accuracy while dealing with different road scenes.

Keywords: vehicle detection; different color modes; road background; vehicle counting; primary and secondary loops

1. Introduction

In recent years, real-time traffic monitoring has attracted extensive attention [1] with the development of intelligent transportation systems (ITSs). In order to build a powerful and reliable intelligent transportation system, vehicle detection and counting is one of the most important parts of collecting and analyzing large mode traffic information data [2]. In particular, it plays an important role in many practical situations, such as solving traffic congestion problems and improving traffic safety.

In the past, vehicle counting was mostly based on a dedicated hardware [3], such as an infrared detector, an electromagnetic induction loop coil, an ultrasonic detector, a radar detector, etc. However, there are many disadvantages of traditional detectors, such as inconvenient installation, maintenance, sensitivity to changes in the environments, etc. [4]. At the same time, the information obtained with traditional technology lacks diversity.

Compared with traditional vehicle-counting technology, counting vehicles based on traffic videos shows great advantages. The video-based method is based on complex image-processing techniques over software. Compared with traditional detectors, a traffic surveillance camera is more convenient for installation and maintenance and has a lower cost. More importantly, the information obtained with traffic videos can be viewed in real-time and is more efficient and flexible. In addition to that, more diverse traffic information could be obtained, which is why it has become an important area of research.

In this paper, a multi-loop vehicle-counting method under gray mode and RGB mode is proposed. To be specific, the preliminary vehicle detection is operated under gray mode
first, and further detection is operated under RGB mode, which could greatly improve the integrity of vehicle detection. Then, the detection information is converted into counting information based on multiple loops to better deal with different influencing factors, such as driving behavior, traffic environment, shooting angle, etc. The counting information is produced per lane.

The rest of the paper is structured as follows: Section 2 introduces related work in vehicle counting. In Section 3, we present a vehicle-detection method under gray mode and RGB mode to improve the integrity of vehicle, and a multi-loop vehicle-counting method to count vehicles with a higher accuracy. The experimental results obtained with the proposed method are presented in Section 4, while a discussion is given in Section 5. Finally, the conclusion is provided in Section 6.

2. Related Work

To obtain counting information, most approaches often need two links: vehicle detection and vehicle counting. For vehicle detection, two kinds of techniques are most often applied: feature-based methods and motion-based methods. Feature-based methods are based on the visual features of a vehicle [5], such as color [6]; texture [7]; edge [8]; or characteristic parts of a vehicle, such as vehicle lights [9], license plates [10], windshield [11], etc. Extraction with low-dimension features is fast and convenient, but it cannot represent all useful information efficiently. In this case, deep learning methods were proposed. These methods need to collect a lot of information prior to training the neural network and then use the trained neural network model to recognize the target. Many deep learning models could help improve the accuracy by a lot. Two-step methods include Spatial Pyramid Pooling Network (SPPNet) [12], Region-Based Convolutional Neural Network (R-CNN) [13], Fast Region-Based Convolutional Neural Network (Fast R-CNN) [14], Faster Region-Based Convolutional Neural Network (Faster R-CNN) [15], etc., while one-step methods include You Only Look Once (YOLO) [16], Single Shot Multi-Box Detector (SSD) [17], Retina-Net [18], SqueezeDet [19], etc. Although the deep learning method has a high accuracy, it is realized at the cost of time-consuming network training by applying a large amount of prior knowledge. Meanwhile, the generality of the model is not very high. The effectiveness of the algorithm may be greatly reduced when the scene is changed. Although transfer learning is also a common way to compensate for this, the effect is often not very good.

Motion-based methods are methods based on the relationship between image sequences. The principle of a motion-based method is to use the correlation between images to segment the moving vehicle from the background, and the video meets the requirements of this method perfectly for its high image refresh rate. Moreover, the motion-based method does not require much prior knowledge but based on a set of images. Therefore, its algorithm complexity is lower and its generality is high. The interframe differencing and background subtraction are the most two characteristic examples. Interframe differencing [20] is based on the difference between two or more successive image frames, whereas background subtraction is based on the difference between moving vehicles and a stationary background [3]. However, due to the influence of environment, the vehicle detected by this method is often incomplete, which further affects the accuracy of vehicle counting. More importantly, most motion-based methods were only applied under gray mode, which is a waste for a color image.

For the counting step, the use of vehicle tracking is a relatively mature approach, such as using a Kalman filter [21] or exploiting feature tracking [22], but its algorithm complexity is high. By contrast, the method based on ROIs has little computational cost. The mechanism of this algorithm is to extract the detection information of the vehicle by analyzing the changes of pixel value in the ROIs and by turning it into the counting information of the vehicle. For example, a linear ROI [23] or a large loop set perpendicular to the road direction could be used to count the total number of vehicles in one direction. To count vehicles by lane, short line or small loop sets for each lane [24] could be applied.
However, when the vehicle does not drive by strictly following the lane, counting based on the ROIs set by each lane could be problematic, such as repeat counts in adjacent lanes or missing counts.

Based on the shortcomings introduced above, a vehicle detection and counting method is proposed. The detection of moving vehicles is based on its motion characteristics and is operated under both gray mode and RGB mode, which could greatly improve the integrity of the detected vehicle. The counting is based on multiple loops with overlapping areas to improve the accuracy of counting. Based on the changes of the pixel value in each loop, the detection information could be converted into lane-splitting counting information.

3. Methods

The proposed method is based on surveillance traffic video. It contains two main parts: vehicle detection and vehicle counting, which are described in the following sections. The block diagram of proposed method is shown in Figure 1.

![Block diagram of the proposed method.](image)

**Figure 1.** Block diagram of the proposed method.

### 3.1. Vehicle Detection under Gray and RGB Modes

As mentioned in Section 2, the counting information is obtained based on the changes of pixel value in multiple loops. Therefore, for vehicle detection, the goal is to extract the vehicle as complete as possible. In view of the problem that the vehicle extracted by the existing motion-based method easily produces holes, a detection method operated under two modes of color, that is gray mode and RGB mode, is proposed to improve the integrity of vehicle.

#### 3.1.1. Preliminary Detection under Gray Mode

The traditional gray background subtraction was applied for preliminary vehicle detection. Under gray mode, there are two main approaches to obtain a gray-mode background: Gaussian mixture model (GMM) [25] and statistical median model (SMM) [26].
GMM has relatively higher extraction accuracy than SMM, but its algorithm complexity is higher and takes 40 to 50 times longer than SMM. Moreover, GMM contains not only the parameter of total number of images in the image library for extracting the gray-mode background \((N)\) but also one more parameter: the learning constant \((\alpha)\), which is an empirical value and is not easily determined [27]. In contrast, SMM contains only one parameter \((N)\). Taking the above factors into consideration, SMM was selected for extracting the gray-mode background in this study.

An image library under gray mode for extracting the gray-mode background should be built first. In this image library, every image is labeled, which could be expressed as \(I_{\text{gray}}^{0}\), where \(n_0 = 1, 2, \ldots, N\), and \(N\) is the total number of images in the image library.

The establishment of background is calculated at the pixel level. The basic idea of establishing the background is to find the gray value in the median position at each pixel in the image library. Different from the previous methods, we not only need to obtain the gray value in the median position at each pixel but also need to record the label of the image that provides the median-position gray value.

At each pixel \((i, j)\), the images in the image library is sorted according to the gray-mode value first:

\[
I_{\text{gray}}^{n_0}(i, j) \leq \cdots \leq I_{\text{gray}}^{n_{(i,j)}}(i, j) \leq \cdots \leq I_{\text{gray}}^{N}(i, j)
\]

where \(n_{(i,j)}\) is the sorted label and \(n_{(i,j)} = 1, 2, \ldots, N\); \(I_{\text{gray}}^{n_{(i,j)}}(i, j)\) is the gray-mode value at each pixel \((i, j)\) of image \(n_{(i,j)}\). In this case, the label of the image that provides the median-position gray value is recorded as follows:

\[
n_{\text{mid}}^{n_{(i,j)}} = \lfloor \frac{N}{2} \rfloor_{(i,j)}
\]

where \(\lfloor x \rfloor\) is the integral function. At each pixel \((i, j)\), the background value equals to the median-position gray value:

\[
B_{\text{gray}}(i, j) = I_{\text{gray}}^{n_{\text{mid}}^{n_{(i,j)}}}(i, j)
\]

where \(B_{\text{gray}}\) is the gray-mode background.

The result of vehicle detection is in the form of binary image, that is, white represents the vehicle and black represents the background. By subtracting the background image from the current frame and by binarizing the subtracting result, vehicle segmentation could be realized. The result of subtracting the background from current frame under gray mode could be expressed as follows:

\[
\delta_{k}^{\text{gray}} = |I_{k}^{\text{gray}} - B_{\text{gray}}|
\]

where \(I_{k}^{\text{gray}}\) is the gray-mode image of the current frame \(k\). The Otsu’s method [28] was used to binarize the subtracting result in this study and the result of vehicle detection under gray mode could be expressed as \(I_{k}^{\text{binarygray}}\).

To illustrate the proposed detection method better, we selected a benchmark dataset for example. The video is called M-30 and could be obtained from the Road-Traffic Monitoring (GRAM-RTM) dataset [29], and a frame \((k = 539)\) from M-30 was selected as the current image for further illustration. The current image under RGB mode \((I_{539}^{RGB})\) and under gray mode \((I_{539}^{\text{gray}})\) are shown in Figure 2a,b. Under the proposed detection method, the established gray-mode background \((B_{\text{gray}})\) is shown in Figure 3a, and the detection under gray mode \((I_{539}^{\text{binarygray}})\) is shown in Figure 3b.
Figure 2. Current frame under (a) RGB mode ($I_{RGB}^{539}$) and (b) gray mode ($I_{RGB}^{539}$).

Figure 3. Detection under gray mode. (a) Established gray-mode background ($B_{gray}^{gray}$). (b) Detection result under gray mode ($I_{binary_{gray}}^{539}$).

As shown in Figure 3b, the detection under gray mode was incomplete, especially for the red vehicle on the right. The reason for this phenomenon is that the background subtraction values at the position of the red vehicle is too small under gray mode. To analyze the unsuccessful detection more clearly, the result of background subtraction under gray mode ($\delta_{k}^{gray}$) was studied by mapping it to another color pattern to make it more intuitive.

The constructed color pattern is a smooth transition from black to red, orange to yellow, and finally to white, as shown in Figure 4. The darker the color, the lower the background subtraction value. By comparing with Figure 2a, it can be seen that the background subtraction values at the position of the red vehicle were generally low, which directly led to the holes in Figure 3b. Therefore, the main reason for this result is that the detection fails when the gray-mode values of foreground and background are close to each other, as could be clearly reflected by comparing Figure 3a with Figure 2b.

Figure 4. Mapped gray-mode result of background subtraction.
In this case, under gray mode, the detection could be said to be a failure, especially for the red vehicle. Under the RGB mode, however, things are different. Next, the detection of vehicle is further operated under RGB mode.

3.1.2. Further Detection under RGB Mode

Under RGB mode, the background subtraction was also applied for vehicle detection. However, there are not many existing methods for the extraction of the RGB background. Here, a method to quickly establish the RGB-mode background based on gray-mode background was proposed.

Since the pixel value of RGB image is composed of three-color channels, namely the red channel, the green channel, and the blue channel, vehicle detection is also realized based on the three-color channels. The current images under red mode \( I_{\text{red}}^{539} \), green mode \( I_{\text{green}}^{539} \), and blue mode \( I_{\text{blue}}^{539} \) are shown in Figure 5a–c. It is worth noting that the image library for extracting the RGB-mode background is the same as that for extracting gray-mode background except that the images in the image library are under red mode, green mode, and blue mode.

![Figure 5](image)

Figure 5. Current frame under (a) red mode \( I_{\text{red}}^{539} \), (b) green mode \( I_{\text{green}}^{539} \), and (c) blue mode \( I_{\text{blue}}^{539} \).

Similarly, the RGB background is also calculated at the pixel level but in a different way. For the gray-mode background, it is established by finding the gray value in the median position at each pixel in the image library and by selecting the gray value as the background value. Additionally, the label of the image that provides the median-position gray value \( n_{\text{mid}}^{(i,j)} \) was recorded, as shown in Equation (2). However, for the RGB-mode background, there is no step to find the value in the median position at each pixel in the image library. Instead, the variable \( n_{\text{mid}}^{(i,j)} \) is used to obtain the background values directly. This is not only to reduce the complexity of the algorithm but also to unify the three-color channels and to reconstruct the RGB background. The background value under red mode, green mode, and blue mode could be expressed as follows:

\[
B_{\text{red}}(i,j) = \frac{I_{\text{red}}^{\text{mid}}(i,j)}{n_{\text{red}}^{(i,j)}}
\]

\[
B_{\text{green}}(i,j) = \frac{I_{\text{green}}^{\text{mid}}(i,j)}{n_{\text{green}}^{(i,j)}}
\]

\[
B_{\text{blue}}(i,j) = \frac{I_{\text{blue}}^{\text{mid}}(i,j)}{n_{\text{blue}}^{(i,j)}}
\]

where \( B_{\text{red}} \), \( B_{\text{green}} \), and \( B_{\text{blue}} \) are the red-mode, green-mode, and blue-mode backgrounds, while \( I_{\text{red}}^{\text{mid}}(i,j) \), \( I_{\text{green}}^{\text{mid}}(i,j) \), and \( I_{\text{blue}}^{\text{mid}}(i,j) \) are the red-mode, green-mode, and blue-mode values at each pixel \( (i,j) \) of image \( I_{\text{RGB}}^{\text{mid}} \). The established background under red mode, green mode, and blue mode are shown in Figure 6a–c. Moreover, with the background under red mode, green mode, and blue mode, the RGB-mode background could be reconstructed, as shown in Figure 6d.
Figure 6. Background under (a) red mode ($B_{\text{red}}$), (b) green mode ($B_{\text{green}}$), and (c) blue mode ($B_{\text{blue}}$) and (d) the reconstructed RGB-mode background ($B_{\text{RGB}}$).

Similarly, the result of subtracting the background from current frame under red mode, green mode, and blue mode could be expressed as follows:

$$\delta_{\text{red}}^k = |I_{\text{red}}^k - B_{\text{red}}|$$  \hspace{1cm} (8)

$$\delta_{\text{green}}^k = |I_{\text{green}}^k - B_{\text{green}}|$$  \hspace{1cm} (9)

$$\delta_{\text{blue}}^k = |I_{\text{blue}}^k - B_{\text{blue}}|$$  \hspace{1cm} (10)

where $I_{\text{red}}^k$, $I_{\text{green}}^k$, and $I_{\text{blue}}^k$ are the current frame $k$ under red mode, green mode, and blue mode.

The mapped results of the background subtraction under red mode, green mode, and blue mode are shown in Figure 7. By comparing Figure 4, Figure 7a, Figure 7b, and Figure 7c, it could be found that the results of background subtraction under gray mode, red mode, green mode, and blue mode were very different. Under gray mode and blue mode, the background subtraction values of the red vehicle were generally low, while under red mode and green mode, the background subtraction values were much higher, especially under red mode. In contrast, for the black vehicle on the left, the results of background subtraction under gray mode were higher.

Figure 7. The mapped result of background subtraction under (a) red mode, (b) green mode, and (c) blue mode.

The vehicle segmentation is obtained by binarizing the subtracting result, as mentioned in Section 3.1.1. In this case, based on subtraction results with different properties, the detection under gray mode, red mode, green mode, and blue mode also has different characteristics, as shown in Figures 3b and 8a–c.

Figure 8. Detection under (a) red mode ($I_{539}^{\text{binary red}}$), (b) green mode ($I_{539}^{\text{binary green}}$), and (c) blue mode ($I_{539}^{\text{binary blue}}$).
3.1.3. Final Detection

As mentioned at the beginning of this section, the goal of vehicle detection is to make the vehicle as complete as possible, but the four detection results are all very incomplete. Each result, when evaluated separately, can be considered a failure.

However, similar to the situation of the background subtraction results, each of them is different in their incompleteness, as shown in Figures 3b and 8a–c. In particular, the detection results of the red vehicle under red mode and green mode were much better than that under gray mode and blue mode, while the detection results of the black vehicle were better under gray mode. Therefore, by taking the union of four detection results, a more ideal detection result could be obtained. The union \((I_{k}^{\text{binary union}})\) could be expressed as follows:

\[
I_{k}^{\text{binary union}} = I_{k}^{\text{binary gray}} \cup I_{k}^{\text{binary gray}} \cup I_{k}^{\text{binary gray}} \cup I_{k}^{\text{binary gray}}
\]  

(11)

As shown in Figure 9a, the union of four detection results \((I_{k}^{\text{binary union}})\) is much more complete and successful than any of the detection shown in Figures 3b and 8a–c. Finally, a simple morphological closing operation [30] could be used to obtain an optimized final detection \(I_{k}^{\text{binary}}\), as shown in Figure 9.

![Figure 9](image)

**Figure 9.** Final detection. (a) Union of four detection results \((I_{k}^{\text{binary union}})\). (b) Optimized final detection \(I_{k}^{\text{binary}}\).

Moreover, in order to further verify the effectiveness of the detection algorithm for vehicles of different colors, in addition to the red and black vehicles shown in Figure 2a, we also detected vehicles of white, silver, blue, brown, yellow, and green, as shown in Figure 10.

3.2. Vehicle Counting Based on Multiple Loops

As can be seen in Figure 9b, the vehicle has been successfully detected with a relatively high degree of integrity. However, it is not enough to obtain counting information. A method to convert detection information into counting information is needed. In this paper, multiple loops with overlapping areas are set to count vehicles by lane. Two kinds of loop, that is the primary loop and secondary loop, are designed under RGB mode first, including its scope, type, number, width, and length, and the designed loops are set on the result of binary detection \(I_{k}^{\text{binary}}\) obtained from Section 3.1. By analyzing the changes of pixel value in each loop, the states of each loop could be judged. Then, based on the state of the loops associated with this lane and adjacent lanes, the occupation of each lane could be determined. Finally, with a strategy set for counting, the lane-splitting counting message could be obtained.
Figure 10. Detection for vehicles of different colors. (a) White vehicle. (b) Silver vehicle. (c) Blue vehicle. (d) Brown vehicle. (e) Yellow vehicle. (f) Green vehicle.

3.2.1. Design of Multiple Loops

As explained in Section 2, in order to count vehicles by lane, multiple short lanes or multiple detection areas are often set up according to lane. However, in actual traffic scenarios, drivers do not always drive strictly following the lanes. In this case, the vehicle may be double counted by two adjacent loops at the same time or missed, especially if the vehicle is right in the middle of two adjacent lanes. To solve this problem, multiple loops that partially overlap each other are set to improve the counting accuracy.

- **Scope of Loops**

While the vehicle may not strictly follow the lane, it is always on the road. Therefore, the basic principle of designing multiple loops is according to the movement of a vehicle on the road. In general, a road is made up of the pavement and shoulders on both sides. The pavement is the part of the road for vehicle driving and is usually divided into lanes, while the shoulder refers to a strip of certain width located from the outer edge of the pavement to the edge of the roadbed, which is used for maintaining the function of the road and temporary parking and serves as lateral support for the pavement, as shown in Figure 11. For the most part, vehicles are on the pavement, but occasionally, they may be on the shoulder of the road. Therefore, in this paper, the overall scope of loops includes both the pavement and shoulder.

Figure 11. Composition of a road.
• **Type and Number of Loops**

  In most studies, only one type of loop was set, which is usually set matching each lane and isolated from each other. However, the loop matching with each lane is sometimes not sufficient for the actual driving situation.

  Different from previous methods, multiple loops that partially overlap each other were designed for counting. Two types of loops were set up in this study, which are called primary loops and secondary loops. For the primary loop, it was set matching each lane, and the range of the set was the pavement. In contrast, for the secondary loop, it was set at the base of the primary loop, and the range of the set concluded at the pavement and shoulder. To be specific, each primary loop matched two secondary loops and two adjacent primary loops shared one secondary loop. Therefore, correspondingly, when the matched primary loop was in the middle of the pavement, the secondary loop matched the two primary loops. When the matched primary loop was located on either side of the pavement, the secondary loop far from the center of the pavement matched only one primary loop. Additionally, it is worth noting that primary loops overlap their matching secondary loops but primary loops or secondary loops themselves do not overlap with each other. In this case, for one lane, one primary loop and two secondary loops are used for counting. Take the road in video M-30 as an example. The road in video M-30 is a four-lane road; in this case, four primary loops and five secondary loops are set, as shown in Figure 12.

![Figure 12](image)

Figure 12. Setting of loops. (a) Primary loops. (b) Secondary loops. (c) The overlap between primary loops and secondary loops.

Moreover, for the sake of description, each lane is expressed as Lane\(^i\), and the number of lane on the road is expressed as \(N_{lane}\). Each primary loop is expressed as PLoop\(^i\), where \(i = 1, 2, \cdots, N_{lane}\), and each secondary loop is expressed as SLoop\(^i\), where \(i = 1, 2, \cdots, N_{lane}, N_{lane} + 1\). In this case, for Lane\(^i\), the three corresponding loops are PLoop\(^i\), SLoop\(^i\) and SLoop\(^{i+1}\).

• **Width of Loops**

  The width of the primary loop is relatively simple to set, which is equal to the width of its corresponding lane, as shown in Figure 12a. For secondary loop, it is a little bit more complicated, depending on the number of the primary loop it matches. When a secondary loop matches two primary loops, its width is half the sum of the two primary loops. When a secondary loop matches only one primary loop, its width is half the width of the primary loop plus the width of the shoulder on the side on which it sits, as shown in Figure 12b.

• **Length of Loops**

  For safety reasons, the front and back vehicles driving on the same lane always keep a certain distance between them. Therefore, if the length of each loop is lower than the safety spacing of vehicles, it is guaranteed that only one vehicle exists in a loop at the same time.

  The safety spacing of vehicles consists of two parts, namely reaction distance and braking distance [31]. Therefore, the length is designed to be less than the reaction distance in this study. The reaction distance is the product of reaction time and driving speed [31]. For reaction time, the American Association of State Highway and Transportation Officials (AASHTO) mandated the use of 2.5 s for the reaction time [32]. While for the driving speed, we set it as 10 km/h—which is a very low speed—to ensure a long enough distance to
react. In this case, the length is designed to be less than seven meters, which can ensure that there is at most one vehicle in $PLoop^i$ or $SLoop^i$.

3.2.2. Judgement of Loop States

The first step to counting the vehicles is to judge the state of each loop, and the designed loops are set on the result of binary detection $I^\text{binary}_k$. As shown in Figure 9b, when a pixel position is in the foreground, the pixel value at that position is 1; otherwise, it is 0. Therefore, the judgement of loop states can be realized by analyzing the change of pixel value in the position of each loop. The binary detection $I^\text{binary}_k$ set with loops is shown in Figure 13.

![Figure 13. Binary detection $I^\text{binary}_k$ set with (a) primary loops and (b) secondary loops.](image)

For each $PLoop^i$ and $SLoop^i$ at frame $k$, we set up a state detector to record the state of $PState^i_k$ and $SState^i_k$, respectively, where $k$ is the frame number of current image. The study on the state of primary loop and secondary loop is similar. For the convenience of description, in the subsequent paragraphs of this subsection, we collectively refer to them as $Loop^i$ and $State^i_k$.

The state of $Loop^i$ was divided into active state and inactive state, representing the state with and without vehicles in $Loop^i$, and the corresponding values are 1 and 0, respectively. As mentioned above, the pixel value at each loop position is the key to determining the state of the loop and we designed three indicators to make the judgement.

The first two are the ratio of the number of pixels with a value of 1 to the total number of pixels contained in the upper line and the lower line, which could be expressed as $upper^i_k$ and $lower^i_k$. When a vehicle enters or leaves one loop, the values of these two indicators change. However, it is not enough to only take the upper line and the lower line of $Loop^i$ as the discriminant basis because some small noises may also cause $upper^i_k$ or $lower^i_k$ to be large. Therefore, the situation of pixel values on the whole region of loop should also be used as the discriminant basis, which could be expressed as $region^i_k$.

When there are no vehicles or noise in $Loop^i$, $upper^i_k$, $lower^i_k$, and $region^i_k$ are all zero. When there is noise in $Loop^i$ but no vehicle, $region^i_k$, $upper^i_k$, or $lower^i_k$ are small. However, $region^i_k$, $upper^i_k$ or $lower^i_k$ are large when there is a vehicle in $Loop^i$. In this case, the state of $Loop^i$ in frame $k$ could be judged based on the three indicators, which could be expressed as follows:

$$State^i_k = \begin{cases} 
1, & \text{if } (upper^i_k > T_{upper}) \lor (lower^i_k > T_{lower}) \land (region^i_k > T_{region}) \\
0, & \text{others}
\end{cases}$$

where $T_{region}$, $T_{upper}$ and $T_{lower}$ are the thresholds for the whole region, the upper line, and the lower line. Additionally, it is worth noting that the activation threshold of the secondary loop is slightly higher than that of the primary loop.
As shown in Figure 13, the state of $PLoop_1$, $SLoop_1$ and $SLoop_2$ are active while the state of other primary loops and secondary loops are inactive. The mathematical form of the states in frame 539 is expressed as Table 1.

### Table 1. Mathematical form of states in frame 539.

| $PState_{539}$ | $SLoop_1$ | $PLoop_1$ | $SLoop_2$ | $SLoop_3$ | $PLoop_3$ | $SLoop_4$ | $PLoop_4$ | $SLoop_5$ |
|---------------|-----------|-----------|-----------|-----------|-----------|-----------|-----------|-----------|
| 1             | 1         | 1         | 0         | 0         | 0         | 0         | 0         | 0         |

#### 3.2.3. Occupancy of Lanes

Whether a Lane is occupied depends not only on the three states ($PLoop_i$, $SLoop_i$, and $SLoop_{i+1}$) corresponding to the lane but also on the loop states of its two adjacent lanes (Lane$^{i-1}$ and Lane$^{i+1}$). Since the lane located on both sides of the road has only one adjacent lane, the situation is relatively simple, so we discuss this type of lane first. Take Lane$^1$ as an example.

The loops corresponding to Lane$^1$ are $SLoop_1$, $PLoop_1$, and $SLoop_2$. However, $SLoop_2$ also corresponds to Lane$^2$. In this case, the primary loop of Lane$^2$ ($PLoop_2$) should also be considered to determine whether Lane$^1$ is occupied. First, we assess $SLoop_1$. It is located at the most edge of the lane, so $SState_1$ is only related to Lane$^1$. In this case, Lane$^1$ can be considered occupied when $SLoop_1$ is activated. Then, we assess $PLoop_1$. It is a kind of primary loop, which is set matching Lane$^1$. Therefore, Lane$^1$ can also be considered occupied when $PLoop_1$ is activated. However, for $SLoop_2$, it is a little complicated because both Lane$^1$ and Lane$^2$ are involved. At this point, we need to consider the status of $PLoop_2$. When the status of $SLoop_2$ is active while the statuses of $SLoop_1$, $PLoop_1$, and $PLoop_2$ are inactive, Lane$^1$ can be considered occupied. However, if the status of $PLoop_2$ is active, Lane$^1$ is not considered occupied. Whether Lane$^1$ is occupied can be summarized in Table 2. A similar analysis could also apply to Lane$^4$, as shown in Table 3.

### Table 2. Judgement of whether Lane$^1$ is occupied.

| $SLoop_1$ | $PLoop_1$ | $SLoop_2$ | $PLoop_2$ | $SLoop_3$ | $PLoop_3$ | $SLoop_4$ | $PLoop_4$ | $SLoop_5$ | Judgement |
|-----------|-----------|-----------|-----------|-----------|-----------|-----------|-----------|-----------|-----------|
| 1         | 1         | 1         | /         | /         | /         | /         | /         | /         | Occupied  |
| 1         | 1         | 0         | /         | /         | /         | /         | /         | /         | Occupied  |
| 1         | 0         | 1         | /         | /         | /         | /         | /         | /         | Occupied  |
| 1         | 0         | 0         | /         | /         | /         | /         | /         | /         | Occupied  |
| 0         | 1         | 1         | /         | /         | /         | /         | /         | /         | Occupied  |
| 0         | 1         | 0         | /         | /         | /         | /         | /         | /         | Occupied  |
| 0         | 0         | 1         | 0         | /         | /         | /         | /         | /         | Occupied  |
| 0         | 0         | 1         | 1         | /         | /         | /         | /         | /         | Not occupied |
| 0         | 0         | 0         | /         | /         | /         | /         | /         | /         | Not occupied |

### Table 3. Judgement of whether Lane$^4$ is occupied.

| $SLoop_1$ | $PLoop_1$ | $SLoop_2$ | $PLoop_2$ | $SLoop_3$ | $PLoop_3$ | $SLoop_4$ | $PLoop_4$ | $SLoop_5$ | Judgement |
|-----------|-----------|-----------|-----------|-----------|-----------|-----------|-----------|-----------|-----------|
| /         | /         | /         | /         | /         | /         | 1         | 1         | 1         | Occupied  |
| /         | /         | /         | /         | /         | /         | 0         | 1         | 1         | Occupied  |
| /         | /         | /         | /         | /         | /         | 1         | 0         | 1         | Occupied  |
| /         | /         | /         | /         | /         | /         | 0         | 0         | 1         | Occupied  |
| /         | /         | /         | /         | /         | /         | 1         | 1         | 0         | Occupied  |
| /         | /         | /         | /         | /         | /         | 0         | 1         | 0         | Occupied  |
| /         | /         | /         | /         | /         | /         | 0         | 1         | 0         | Occupied  |
| /         | /         | /         | /         | /         | /         | 1         | 1         | 0         | Not occupied |
| /         | /         | /         | /         | /         | /         | 0         | 0         | 0         | Not occupied |
Now, regarding the lane located on the middle of the road, this kind of lane has two adjacent lanes, and the situation is relatively complicated. Here, take Lane³ as an example. The loops corresponding to Lane³ are SLoop³, PLoop³, and SLoop⁴. However, SLoop³ also corresponds to Lane², while SLoop⁴ also corresponds to Lane⁴. In this case, the primary loop of Lane² (PLoop²) and Lane⁴ (PLoop⁴) should also be considered to determine whether Lane³ is occupied. First, we assess PLoop³. It is set to match Lane³, so Lane³ can be considered occupied when PLoop³ is activated. For SLoop³, when the status of SLoop³ is active while the statuses of PLoop², PLoop³, and SLoop⁴ are inactive, Lane¹ can be considered occupied. However, if the status of PLoop² is active, Lane³ should not be considered occupied. For SLoop⁴, when the status of SLoop⁴ is active while the status of PLoop³ PLoop⁴ is inactive, Lane³ can be considered occupied. However, if the status of PLoop⁴ is active, Lane³ should not be considered occupied. Whether Lane¹ is occupied can be summarized in Table 4. A similar analysis could also apply to Lane², as shown in Table 5.

### Table 4. Judgement of whether Lane³ is occupied.

| SLoop¹ | PLoop¹ | SLoop² | PLoop² | SLoop³ | PLoop³ | SLoop⁴ | PLoop⁴ | SLoop⁵ | Judgement |
|--------|--------|--------|--------|--------|--------|--------|--------|--------|-----------|
| /      | /      | /      | /      | 1      | 1      | 1      | /      | /      | Occupied  |
| /      | /      | /      | /      | 0      | 1      | 1      | /      | /      | Occupied  |
| /      | /      | /      | /      | 1      | 0      | 1      | 0      | /      | Occupied  |
| /      | /      | /      | /      | 1      | 0      | 1      | 1      | /      | Not occupied |
| /      | /      | /      | /      | 0      | 0      | 1      | 0      | /      | Occupied  |
| /      | /      | /      | /      | 1      | 1      | 0      | /      | /      | Occupied  |
| /      | /      | /      | /      | 0      | 1      | 0      | /      | /      | Occupied  |
| /      | /      | /      | /      | 0      | 1      | 0      | 0      | /      | Occupied  |
| /      | /      | /      | /      | 1      | 1      | 0      | 0      | /      | Not occupied |
| /      | /      | /      | /      | 0      | 0      | 0      | 0      | /      | Not occupied |

### Table 5. Judgement of whether Lane² is occupied.

| SLoop¹ | PLoop¹ | SLoop² | PLoop² | SLoop³ | PLoop³ | SLoop⁴ | PLoop⁴ | SLoop⁵ | Judgement |
|--------|--------|--------|--------|--------|--------|--------|--------|--------|-----------|
| /      | /      | 1      | 1      | 1      | /      | /      | /      | /      | Occupied  |
| /      | /      | 1      | 1      | 0      | /      | /      | /      | /      | Occupied  |
| /      | 0      | 1      | 0      | 1      | /      | /      | /      | /      | Occupied  |
| /      | 1      | 1      | 0      | 1      | /      | /      | /      | /      | Not occupied |
| /      | 0      | 1      | 0      | 0      | /      | /      | /      | /      | Occupied  |
| /      | 1      | 1      | 0      | 0      | /      | /      | /      | /      | Not occupied |
| /      | 0      | 1      | 1      | 1      | /      | /      | /      | /      | Occupied  |
| /      | 0      | 0      | 1      | 0      | /      | /      | /      | /      | Occupied  |
| /      | 0      | 0      | 1      | 1      | /      | /      | /      | /      | Not occupied |
| /      | 0      | 0      | 0      | 0      | /      | /      | /      | /      | Not occupied |

Using the above criterion, the occupancy of each lane shown in Figure 13 and Table 1 can be discriminated, that is Lane¹ is occupied while Lane², Lane³, and Lane⁴ are not occupied.

3.2.4. Counting Strategy

To count vehicles, we set up a counter variable, which is used to count the cumulative number of vehicles in each lane. Slightly different from the setting of the loop, for each lane, only one counter that matches the lane is set and no secondary counters are set. The counter could be expressed as Countedᵢ, where i = 1, 2, · · · , Nₙ, which represents the cumulative number of vehicles driving on Laneᵢ that have been counted until frame k.
The strategy of increasing the value of \( \text{Counted}_k \) is as follows: if and only if the lane is not occupied by a vehicle in the previous frame and is occupied in the current frame, the cumulative number of vehicles on \( \text{Lane}^i \) should be increase by 1; otherwise, the cumulative number remains the same. In this case, the cumulative number of vehicles driving on \( \text{Lane}^i \) until frame \( k \) could be expressed as follows:

\[
\text{Counted}^i_k = \begin{cases} 
\text{Counted}^i_{k-1} + 1, & \text{if \( \text{Lane}^i \) is not occupied at \( k-1 \) but occupied at \( k \)} \\
\text{Counted}^i_{k-1}, & \text{otherwise}
\end{cases}
\]  

(13)

As can be seen from Equation (13), in order to accurately count the vehicles on the lane, two important nodes need to be accurately discriminated. The first one is from the frame in which the lane is not occupied to the frame in which the lane is occupied. At this point, the counter for the lane increases by 1 in the frame in which the lane is occupied. The second important node is from the frame in which the lane is occupied to the frame in which the lane is not occupied. This is in preparation for the next increment in the counter. Therefore, these four frames are key for the counter to count properly.

Take the left black vehicle in Figure 2a as an example. The loop states of its four key frames is expressed in Table 6 and displayed in Figure 14. Using the counting strategy described above, the counter for \( \text{Lane}^1 \) should be increased by 1 in frame 534.

Table 6. The state of loops at the four key frames of the left black vehicle in Figure 2a.

| SLoop | PLoop | SLoop | PLoop | SLoop | PLoop | SLoop | PLoop | Occupancy of Lane^1 |
|-------|-------|-------|-------|-------|-------|-------|-------|---------------------|
| Frame 533 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | Not occupied |
| Frame 534 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | Occupied |
| Frame 544 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | Occupied |
| Frame 545 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | Not occupied |

Figure 14. Four key frames of the left black vehicle in Figure 2a. (a) Frame 533 (\( \text{Lane}^1 \) is not occupied). (b) Frame 534 (\( \text{Lane}^1 \) is occupied). (c) Frame 544 (\( \text{Lane}^1 \) is occupied). (d) Frame 545 (\( \text{Lane}^1 \) is not occupied).

3.3. Evaluation Index

To evaluate the counting result, four metrics were used in this paper, that is, \( \text{Recall} \), \( \text{Precision} \), \( \text{F-measure} \), and \( \text{Accuracy} \).

\( \text{Recall} \) is a measure of the success in detecting relevant objects, that is, the percentage of vehicles correctly counted in all vehicles that should be counted, while \( \text{Precision} \) is the percentage of vehicles correctly counted in all counted objects. These two metrics could be expressed as follows [33]:

\[
\text{Recall} = \frac{TP}{TP + FN}
\]  

(14)
Precision = \frac{TP}{TP + FP} \quad (15)

where $TP$ is the number of vehicles correctly counted, $FN$ is the number of vehicles that should be counted but not, and $FP$ is the number of objects that should be not counted. In this case, the number of vehicles that should be counted ($True$) and the number of counted objects ($Counted$) could be calculated as follows:

$True = TP + FN$, $Counted = TP + FP \quad (16)$

where $F$-measure is the weighted harmonic average of $Recall$ and $Precision$, which combines the results of $Recall$ and $Precision$, and we make the weights equal to each other in this paper. In this case, $F$-measure could be expressed as follows [33]:

\begin{equation}
F\text{-measure} = 2 \times \frac{Recall \times Precision}{Recall + Precision} \quad (17)
\end{equation}

In contrast, for $Accuracy$, it is used to evaluate the difference between the true value and the counted value, which is defined as follows [34]:

\begin{equation}
Accuracy = 1 - \frac{|True - Counted|}{True} \quad (18)
\end{equation}

Using Equation (16) to replace $True$ and $Counted$ with $TP$, $FN$, and $FP$, Equation (18) could also be expressed as follows:

\begin{equation}
Accuracy = 1 - \frac{|FN - FP|}{TP + FN} \quad (19)
\end{equation}

Let us analyze these metrics briefly. Compared with the $F$-measure introduced above, $Accuracy$ evaluates the overall difference between counted values and true values, ignoring the details of errors in treating noise as vehicle or errors with missing vehicles. As a result, it is a more direct metric of the overall outcome. Therefore, a comprehensive analysis of $F$-measure and $Accuracy$ is more scientific.

### 3.4. Experimental Dataset

In this paper, the performance of the proposed method was experimented on six traffic videos. Video I and Video II are two videos recorded on a highway, which are called M-30 and M-30-HD and could be obtained from the GRAM-RTM dataset [29]. These two videos were shot from the rear of the vehicle on a sunny and cloudy day. Video III is also a video recorded on a highway, which is called Highway and could be obtained from the Change Detection Benchmark [35]. Unlike Video I and Video II, it was shot from the front. The weather of Video III is sunny.

Video IV, Video V, and Video VI are three videos [36] recorded by us in Nanjing, Jiangsu, China, which are called NJ-1, NJ-2, and NJ-3. Similar to Video III, Video IV was also shot from the front, but the location is on an urban road and the weather is cloudy. Video V and Video VI were shot from the side on two expressways on a sunny and cloudy day, respectively.

The basic information in each video and road is shown in Table 7, including the name of video, recording site, number of frames, and the average pixel number for a vehicle. Among them, the meaning of "Average Pixel No. for a Vehicle" needs some explanation. As we all know, the imaging principle of the camera is pinhole imaging. Therefore, through perspective, the same vehicle in the image appears to be small at a distance but large at close distances. Based on this, for the statistics of the pixel number for a vehicle in each video, we aim for the average pixel number of vehicles that are at close distance. At the same time, it is worth mentioning that the multi-loop is also set at close distance. From this point of view, the information of how many pixels for a vehicle is useful to apply the proposed method in different videos. The characteristics in each video and road is shown in Table 8, including road category, number of lanes, weather, and shooting angle.
Table 7. Basic information of the videos and the roads.

| Video Name | Recording Site | Frame No. | Average Pixel No. for a Vehicle |
|------------|----------------|-----------|--------------------------------|
| Video I    | M-30 Madrid, Spain | 7520     | 4000 ~ 5000                     |
| Video II   | M-30-HD Madrid, Spain | 9390     | 4000 ~ 5000                     |
| Video III  | Highway N/A      | 1700     | 4000 ~ 4500                     |
| Video IV   | NJ-1 Jiangsu, China | 10,000   | 13,000 ~ 15,000                 |
| Video V    | NJ-2 Jiangsu, China | 10,000   | 9000 ~ 10,000                   |
| Video VI   | NJ-3 Jiangsu, China | 10,000   | 9000 ~ 10,000                   |

Table 8. Characteristics of the videos and the roads.

| Road Category | Lane No. | Weather | Shooting Angle |
|---------------|----------|---------|----------------|
| Video I       | Highway 4 | Sunny   | Rear           |
| Video II      | Highway 4 | Cloudy  | Rear           |
| Video III     | Highway 2 | Sunny   | Front          |
| Video IV      | Urban Road 3 | Cloudy | Front          |
| Video V       | Expressway 3 | Sunny | Side           |
| Video VI      | Expressway 3 | Cloudy | Side           |

4. Results

The experimental results of each sequence are shown in Tables 9–15, and examples of experimental results of each sequence are shown in Figures 15–20.

Table 9. Experimental results of Video I.

| Lane | TP  | FN  | FP  | True | Counted | Recall | Precision | F-Measure | Accuracy |
|------|-----|-----|-----|------|---------|--------|-----------|-----------|----------|
| 1    | 79  | 0   | 0   | 79   | 79      | 100%   | 100%      | 100%      | 100%     |
| 2    | 70  | 0   | 1   | 70   | 71      | 100%   | 98.59%    | 99.27%    | 98.57%   |
| 3    | 69  | 0   | 1   | 69   | 70      | 100%   | 98.57%    | 99.28%    | 98.55%   |
| 4    | 36  | 0   | 0   | 36   | 36      | 100%   | 100%      | 100%      | 100%     |

Figure 15. Example of the experimental results in Video I.

Table 10. Experimental results of Video II.

| Lane | TP  | FN  | FP  | True | Counted | Recall | Precision | F-Measure | Accuracy |
|------|-----|-----|-----|------|---------|--------|-----------|-----------|----------|
| 1    | 68  | 0   | 1   | 68   | 69      | 100%   | 98.55%    | 99.27%    | 98.53%   |
| 2    | 75  | 0   | 1   | 75   | 76      | 100%   | 98.68%    | 99.34%    | 98.67%   |
| 3    | 59  | 1   | 0   | 60   | 59      | 98.33% | 100%      | 99.16%    | 98.33%   |
| 4    | 31  | 0   | 0   | 31   | 31      | 100%   | 100%      | 100%      | 100%     |
Figure 16. Example of the experimental results in Video II.

Table 11. Experimental results of Video III.

| Lane | TP  | FN  | FP  | True | Counted | Recall | Precision | F-Measure | Accuracy |
|------|-----|-----|-----|------|---------|--------|-----------|-----------|----------|
| 1    | 17  | 0   | 0   | 17   | 17      | 100%   | 100%      | 100%      | 100%     |
| 2    | 10  | 0   | 0   | 10   | 10      | 100%   | 100%      | 100%      | 100%     |

Figure 17. Example of the experimental results in Video III.

Table 12. Experimental results of Video IV.

| Lane | TP  | FN  | FP  | True | Counted | Recall | Precision | F-Measure | Accuracy |
|------|-----|-----|-----|------|---------|--------|-----------|-----------|----------|
| 1    | 4   | 0   | 0   | 4    | 4       | 100%   | 100%      | 100%      | 100%     |
| 2    | 36  | 1   | 1   | 37   | 37      | 97.30% | 97.30%    | 97.30%    | 100%     |
| 3    | 44  | 0   | 1   | 44   | 45      | 100%   | 97.78%    | 98.88%    | 97.73%   |
| 4    | 19  | 1   | 1   | 20   | 20      | 95.00% | 95.00%    | 95.00%    | 100%     |

Figure 18. Example of the experimental results in Video IV.

Table 13. Experimental results of Video V.

| Lane | TP  | FN  | FP  | True | Counted | Recall | Precision | F-Measure | Accuracy |
|------|-----|-----|-----|------|---------|--------|-----------|-----------|----------|
| 1    | 44  | 1   | 1   | 45   | 45      | 97.78% | 97.78%    | 97.78%    | 100%     |
| 2    | 123 | 0   | 2   | 123  | 125     | 100%   | 98.40%    | 99.19%    | 98.37%   |
| 3    | 114 | 1   | 2   | 115  | 116     | 99.13% | 98.28%    | 98.70%    | 99.13%   |
The total number of vehicles that should be counted are 988, and there are only six missing errors and fourteen noise errors. As a whole, the total Recall, Precision, F-measure, and Accuracy of the proposed method achieved 99.39%, 98.59%, 98.99%, and 99.19%, which reached more than 98.5%.

5. Discussion

In this section, we make a comparison with other methods and discuss the factors that affect the experimental results and the countermeasures of the proposed method. Additionally, the limitations of the proposed method are analyzed in Section 5.3.

5.1. Comparison with Recent Methods

The proposed method was compared to state-of-the-art methods that used the three benchmark sequences: Highway (Video I), M-30 (Video II), and M-30-HD (Video III). As shown in Tables 16 and 17, the proposed method achieved the highest accuracy, with no error in the Highway sequence.

For M-30-HD, the accuracy of the proposed method is lower than that in [37]. However, the experimental vehicles in the study are larger, which increases the probability of errors, as shown in Table 18.
Table 16. Comparison results of the benchmark Highway.

|                | True | Counted | Recall  | Precision | F-Measure | Accuracy |
|----------------|------|---------|---------|-----------|-----------|----------|
| Shakeri [38]   | 27   | N/A     | 85.40%  | 95.10%    | 89.99%    | N/A      |
| H. Yang [39]   | 16   | 14      | 88.89%  | 100%      | 94.12%    | 87.50%   |
| Abdelwahab [37]| 27   | 29      | 100%    | 93.10%    | 96.43%    | 92.59%   |
| Gomaa [40]     | 27   | N/A     | 99.20%  | 98.20%    | 98.70%    | N/A      |
| Rosas-Arias [41]|27   | 26      | 96.43%  | 100%      | 98.18%    | 96.30%   |
| Proposed Method| 27   | 27      | 100%    | 100%      | 100%      | 100%     |

Table 17. Comparison results of the benchmark M-30.

|                | True | Counted | Recall  | Precision | F-Measure | Accuracy |
|----------------|------|---------|---------|-----------|-----------|----------|
| Bouvié [42]    | 77   | 69      | 89.61%  | 100%      | 94.52%    | 89.61%   |
| Quesada [34]   | 77   | 75      | 97.40%  | 100%      | 98.68%    | 97.40%   |
| H. Yang [39]   | 77   | 71      | 88.51%  | 95.06%    | 91.67%    | 92.21%   |
| Abdelwahab [37]| 77   | 76      | 98.72%  | 100%      | 99.35%    | 98.70%   |
| Harikrishnan [43]|77   | 76      | N/A     | N/A       | N/A       | 98.70%   |
| Proposed Method| 254  | 256     | 100%    | 99.22%    | 99.61%    | 99.21%   |

Table 18. Comparison results of the benchmark M-30-HD.

|                | True | Counted | Recall  | Precision | F-Measure | Accuracy |
|----------------|------|---------|---------|-----------|-----------|----------|
| Bouvié [42]    | 42   | 33      | 78.57%  | 100%      | 88.00%    | 78.57%   |
| Quesada [34]   | 42   | 39      | 92.86%  | 100%      | 96.30%    | 92.86%   |
| H. Yang [39]   | 42   | 37      | 89.36%  | 100%      | 94.38%    | 88.10%   |
| Abdelwahab [37]| 42   | 42      | 100%    | 100%      | 100%      | 100%     |
| Harikrishnan [43]|42   | 40      | N/A     | N/A       | N/A       | 95.24%   |
| Proposed Method| 234  | 235     | 99.57%  | 99.15%    | 99.36%    | 99.57%   |

5.2. Discussion of Influencing Factors and Countermeasures

Let us give a discussion of the influencing factors from the aspects of the characteristics mentioned in Table 8: road category, number of lanes, weather and shooting angle, and the countermeasures of the proposed method.

- **Road Category**

  Road category is a relatively important factor that has a certain influence on both vehicle detection and vehicle counting. Specifically, its influence on the results is mainly reflected in the complexity of the traffic environment and road environment. Compared with videos taken from the highways (Figures 15–17), the traffic environment and road environment of videos taken from urban roads (Figure 18) and expressways (Figures 19 and 20) are more complex, such as the higher proportion of large vehicles and more trees on both sides of the road. In the same way, videos taken from urban roads may contain more disadvantages than videos taken from expressways.

  Therefore, it is not enough to carry out the experiment only on the highway. We also added three groups of experiments on urban roads and expressways. The influence of the road category could also be reflected in the experiment. As shown in Table 15, the three benchmarks (Video I, Video II, and Video III), which were all taken from highways, achieved the highest accuracy. Although the accuracy of urban roads (Video IV) and expressways (Video V and Video VI) is not as good as that of the highway, it is not bad either.
• **Number of Lanes**

The influence of lane number is mainly reflected in the vehicle count. To be specific, it basically shows the trend that the less lanes, the higher the accuracy of the results. This is mainly because vehicle counting is easily affected by vehicles driving side by side in different lanes. At the same time, as the number of lanes increases, drivers have more choices of roads to take, which makes it easier for drivers to engage in driving behaviors that do not strictly follow the designated lane lines. Therefore, the ability of a traditional ROI completely matching the lane is a little inadequate to deal with this situation. Additionally, this is one of the reasons that multiple loops were set on the road.

As shown in Figure 21a, this is a vehicle driving right on the middle of Lane2 and Lane3. If only four traditional primary loops were set, the counting for this vehicle may be problematic. To be specific, with stricter thresholds, PLoop2 and PLoop3 are not activated and this vehicle is missed. If looser thresholds are set, the PLoop2 and PLoop3 are activated at the same time and this vehicle is counted repeatedly. However, with the multiple loops and counting strategy designed in this study, this vehicle is only counted once on Lane3, regardless of the strict threshold or the loose threshold, as shown in Table 19.

![Figure 21](image)

**Figure 21.** Example of a vehicle driving right in the middle of Lane2 and Lane3. (a) Example frame under RGB mode. (b) Detection result set with primary loops. (c) Detection result set with secondary loops.

|      | PLoop1 | PLoop2 | PLoop3 | PLoop4 | Lane2 | Lane3 |
|------|--------|--------|--------|--------|-------|-------|
| Stricter | 0      | 0      | 0      | 0      | ×     | ×     |
| Looser   | 0      | 1      | 1      | 0      | ✓     | ✓     |

|      | SLoop1 | PLoop1 | SLoop2 | PLoop2 | SLoop3 | PLoop3 | SLoop4 | PLoop4 | SLoop5 | Lane2 | Lane3 |
|------|--------|--------|--------|--------|--------|--------|--------|--------|--------|-------|-------|
| Stricter | 0      | 0      | 0      | 0      | 1      | 0      | 0      | 0      | 0      | ×     | ✓     |
| Looser   | 0      | 0      | 0      | 1      | 1      | 1      | 0      | 0      | 0      | ×     | ✓     |

The influence of lane number could be also reflected in the experimental results. As shown in Table 15, Video III, with only two lanes, achieved the highest accuracy. However, for Video V and Video VI, they only contain three lanes but the experimental results are not as good as Video I and Video II, which contain four lanes. One of the reasons is that Video I and Video II were shot on the highway with a simpler environment, but the setting of multiple loops also helped to greatly improve the accuracy of the experimental results.

Moreover, in terms of where the lane is on the road, errors are more likely to occur in the median lane and its adjacent lanes. For example, errors all occurred in the two median lanes (Lane2 and Lane3) in Video I (Table 9), while errors all occurred in its median lane (Lane2) in Video VI (Table 14). Of course, this also has to do with the general tendency of drivers to drive in the median lane, which further increases the chance of errors. While in Video II, Video IV, and Video V, errors not only occurred in the median lane but also occurred in its adjacent lane. This is especially evident in Video V. As shown in Table 13, two errors occurred in Lane2 and three errors occurred in Lane3. This is mainly due to the phenomenon of vehicles driving side by side on different lanes affecting the vehicle count.
At the same time, vehicles not only tend to drive in Lane\(^2\) but also tend to drive in Lane\(^3\), which is also a factor. This could be seen in the number of vehicles. As shown in Table 13, the number of vehicles that should be counted (True) in Lane\(^2\) is 123, while the True number in Lane\(^2\) is 116, which is also a reflection of the propensity of the driving behavior for drivers. Therefore, compared with the lanes on both sides of the road, the counting strategies proposed in Section 3.2.4 set for the middle lane are also more complex.

- **Weather**

  Weather mainly affects vehicle detection. A video shot in sunny weather is brighter in color. Therefore, the detection algorithm in RGB mode proposed in this study has better ability to repair vehicle holes. In addition, compared with a sunny day, a cloudy day contains more undesirable factors, such as changes in the lighting environment caused by the influence of clouds. As shown in Table 15, the three videos taken on sunny days (Video I, Video III, and Video V) have better performance overall than those taken on cloudy days (Video II, Video IV, and Video VI).

- **Shooting Angle**

  As for the shooting angle, it mainly affects vehicle counting, and it is due to the distortion of the image caused by perspective. The imaging principle of the camera is pinhole imaging. As a result, the view of the camera on the road and the vehicle usually produce a certain distortion. In general, the more oblique the shooting angle, the more severe the distortion. As shown in Figure 16, the projection of vehicle driving on lane\(^1\) was tilted to the left, while the projection of vehicle driving on lane\(^4\) was tilted to the right. This phenomenon is most obvious when the vehicles were shot from the side. As shown in Figures 19 and 20, the projection of vehicle was tilted to the right by very much.

  If only the primary loops were set, there may be some errors in counting. However, with the help of the secondary loops, the counting was not greatly affected by the shooting angle. The experimental results also prove that the multi-loop setup works well against this adverse effect, as shown in Table 15.

### 5.3. Analysis of Limitations of the Proposed Method

Benefiting from vehicle detection under two color modes and the setting with multiple loops, the counting accuracy of vehicles greatly improved, but there are still some false results. There are two types of counting errors, namely missed counting (\(FN\)) and redundant counting (\(FP\)). We start from these two aspects to analyze the causes of the errors and the limitations of the proposed method.

- **Missed Counting**

  Incomplete vehicle detection and unreasonable ROI settings could both lead to the generation of missed counting, while redundant counting is mainly caused by unreasonable ROI settings. Therefore, in previous studies, a reduction in counting accuracy was mainly due to errors missing counts. Benefiting from vehicle detection under two color modes and the setting with multiple loops proposed in this paper, the number of errors due to missing counts was greatly reduced and the counting accuracy significantly improved. However, as shown in Table 15, there are still a small number of missing counts (\(FN = 6\)) in the experiments, which is mainly affected by incomplete detection of the vehicle and has little to do with the setting with multiple loops. Here is an example. As shown in Figure 22a, the color of the incomplete part is very similar to the color of the road background. Therefore, it is difficult to separate them no matter whether under gray mode or RGB mode, resulting in incomplete vehicle detection. As a result, no matter using the primary loops or the secondary loops, this vehicle does not meet the activation conditions, leading to a missed count.
• Redundant Counting

The error of redundant counting mainly occurs for large vehicles, that is, counting are carried out simultaneously in two adjacent lanes. As shown in Figure 23a, this large vehicle spanned two lanes, so it was counted both in lane 1 and lane 2, resulting in redundant counting. This phenomenon could also be proven by the experimental results. As shown in Table 15, the error of redundant counting (FP) is more likely to occur in Video IV, Video V, and Video VI, mainly because the proportion of large vehicles on urban roads and expressways is higher than on highways.

6. Conclusions

In this paper, we presented a multi-loop vehicle counting method under gray mode and RGB mode. For vehicle detection, the detection operated under gray mode and RGB mode and the final detection was obtained by integrating four detection results and by optimizing using a morphological closing operation. Under this kind of algorithm, the integrity of the detected vehicles greatly improved. As for counting, the setting with multiple of loops could count vehicles by lane and could better deal with different influencing factors, such as driving behavior, traffic environment, shooting angle, etc.

The experimental results showed that the proposed method performs well in videos shot in different weather at different road scenes. The proposed method successfully counted the vehicles with a high performance, e.g., the total Recall, Precision, F-measure, and Accuracy of proposed method achieved 99.39%, 98.59%, 98.99%, and 99.19%, all of which are more than 98.5%.

There is still much to study in future work. As analyzed in Section 5.3, when the color of the vehicle is very similar to the color of the road background, the effect of vehicle detection is not ideal. Therefore, it is necessary to find other methods to further improve the detection algorithm, such as the applications in proposed regions with objects to further improve the integrity of the vehicle, etc.

Author Contributions: Conceptualization, Y.C. and J.L.; methodology, Y.C.; software, Y.C.; validation, Y.C.; writing—original draft preparation, Y.C.; writing—review and editing, Y.C. and J.L. Both authors have read and agree to the published version of the manuscript.
Funding: This research was funded by the Jiangsu Provincial Key Research and Development Program (Social Development) Project (BE2018754).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The data presented in this study are available on request from the author.

Conflicts of Interest: The authors declare no conflict of interest.

References
1. Buch, N.; Velastin, S.A.; Orwell, J. A Review of Computer Vision Techniques for the Analysis of Urban Traffic. IEEE Trans. Intell. Transp. Syst. 2011, 12, 920–939. [CrossRef]
2. Tsai, L.W.; Hsieh, J.W.; Fan, K.C. Vehicle detection using normalized color and edge map. IEEE Trans. Image Process. 2007, 16, 850–864. [CrossRef] [PubMed]
3. Celik, T.; Kusetogullari, H. Solar-Powered Automated Road Surveillance System for Speed Violation Detection. IEEE Trans. Ind. Electron. 2010, 57, 3216–3227. [CrossRef]
4. Yang, Z.; Pun-Cheng, L.S.C. Vehicle detection in intelligent transportation systems and its applications under varying environments: A review. Image Vis. Comput. 2018, 69, 143–154. [CrossRef]
5. Liu, Y.; Tian, B.; Chen, S.; Zhu, F.; Wang, K. A survey of vision-based vehicle detection and tracking techniques in ITS. In Proceedings of the 2013 IEEE International Conference on Vehicular Electronics and Safety (ICVES 2013), Dongguan, China, 28–30 July 2013; pp. 72–77. [CrossRef]
6. Yang, Y.; Gao, X.; Yang, G. Study the method of vehicle license locating based on color segmentation. In Proceedings of the 2011 International Conference on Advanced in Control Engineering and Information Science (CIES 2011), Dali, China, 18–19 August 2011; Volume 15, pp. 1324–1329. [CrossRef]
7. Haralick, R.; Shanmugam, K.; Dinstein, I. Textural features for image classification. IEEE Trans. Syst. Man Cybern. 1973, smc 3, 610–621. [CrossRef]
8. Matthews, N.; An, P.; Charnley, D.; Harris, C. Vehicle detection and recognition in greyscale imagery. Control Eng. Pract. 1996, 4, 473–479. [CrossRef]
9. Chen, D.Y.; Lin, Y.H.; Peng, Y.J. Nighttime brake-light detection by Nakagami imaging. IEEE Trans. Intell. Transp. Syst. 2012, 13, 1627–1637. [CrossRef]
10. Abolghasemi, V.; Ahmadyfard, A. An edge-based color-aided method for license plate detection. Image Vis. Comput. 2009, 27, 1134–1142. [CrossRef]
11. Yang, J.; Wang, Y.; Sowmya, A.; Li, Z. Vehicle detection and tracking with low-angle cameras. In Proceedings of the 2010 IEEE International Conference on Image Processing, ICIP 2010, Hong Kong, China, 26–29 September 2010; pp. 685–688. [CrossRef]
12. He, K.; Zhang, X.; Ren, S.; Sun, J. Spatial pyramid pooling in deep convolutional networks for visual recognition. In Proceedings of the 13th European Conference on Computer Vision, ECCV 2014, Zurich, Switzerland, 6–12 September 2014; Volume 8691 LNCS, pp. 346–361. [CrossRef]
13. Girshick, R.; Donahue, J.; Darrell, T.; Malik, J. Rich feature hierarchies for accurate object detection and semantic segmentation. In Proceedings of the 27th IEEE Conference on Computer Vision and Pattern Recognition (CVPR 2014), Columbus, OH, USA, 23–28 June 2014; pp. 580–587. [CrossRef]
14. Girshick, R. Fast R-CNN. In Proceedings of the 15th IEEE International Conference on Computer Vision, (ICCV 2015), Santiago, Chile, 11–18 December 2015; pp. 1440–1448. [CrossRef]
15. Ren, S.; He, K.; Girshick, R.; Sun, J. Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks. IEEE Trans. Pattern Anal. Mach. Intell. 2017, 39, 1137–1149. [CrossRef] [PubMed]
16. Redmon, J.; Divvala, S.; Girshick, R.; Farhadi, A. You only look once: Unified, real-time object detection. In Proceedings of the 29th IEEE Conference on Computer Vision and Pattern Recognition, (CVPR 2016), Las Vegas, NV, USA, 26 June–1 July 2016; pp. 779–788. [CrossRef]
17. Liu, W.; Anguelov, D.; Erhan, D.; Szegedy, C.; Reed, S.; Fu, C.Y.; Berg, A.C. SSD: Single shot multibox detector. In Proceedings of the 14th European Conference on Computer Vision, (ECCV 2016), Amsterdam, The Netherlands, 8–16 October 2016; Volume 9905 LNCS, pp. 21–37. [CrossRef]
18. Lin, T.Y.; Goyal, P.; Girshick, R.; He, K.; Dollar, P. Focal Loss for Dense Object Detection. In Proceedings of the 16th IEEE International Conference on Computer Vision, (ICCV 2017), Venice, Italy, 22–29 October 2017; pp. 2999–3007. [CrossRef]
19. Wu, B.; Iandola, F.; Jin, P.H.; Keutzer, K. SqueezeDet: Unified, Small, Low Power Fully Convolutional Neural Networks for Real-Time Object Detection for Autonomous Driving. In Proceedings of the 30th IEEE Conference on Computer Vision and Pattern Recognition Workshops, (CVPRW 2017), Honolulu, HI, USA, 21–26 July 2017; pp. 446–454. [CrossRef]
20. Kim, J.B.; Kim, H.J. Efficient region-based motion segmentation for a video monitoring system. Pattern Recognit. Lett. 2003, 24, 113–128. [CrossRef]
21. Zhang, X.; Gao, H.; Xue, C.; Zhao, J.; Liu, Y. Real-time vehicle detection and tracking using improved histogram of gradient features and Kalman filters. *Int. J. Adv. Robot. Syst.* 2018, 15. [CrossRef]

22. Abdelwahab, M.A.; Abdelwahab, M.M. A Novel Algorithm for Vehicle Detection and Tracking in Airborne Videos. In Proceedings of the 17th IEEE International Symposium on Multimedia, (ISM 2015), Miami, FL, USA, 14–16 December 2015; pp. 65–68. [CrossRef]

23. Kadikis, R.; Freivalds, K. Vehicle Classification in Video Using Virtual Detection Lines. In *Sixth International Conference on Machine Vision (icmv 2013)*; Verikas, A., Vuksanovic, B., Zhou, J., Eds.; Spie-Int Soc Optical Engineering: Bellingham, WA, USA, 2013; Volume 9067, p. 906715.

24. Barcellos, F.; Bouvie, C.; Escouto, F.L.; Scharcanski, J. A novel video based system for detecting and counting vehicles at user-defined virtual loops. *Expert Syst. Appl.* 2015, 42, 1845–1856. [CrossRef]

25. Stauffer, C.; Grimson, W.E.L. Learning patterns of activity using real-time tracking. *IEEE Trans. Pattern Anal. Mach. Intell.* 2000, 22, 747–757. [CrossRef]

26. Gloyer, B.; Aghajan, H.K.; Siu, K.Y.S.; Kailath, T. Video-based freeway-monitoring system using recursive vehicle tracking. In Proceedings of the Image and Video Processing III, San Jose, CA, USA, 9–10 February 1995; Volume 2421, pp. 173–180.

27. Wan, Q.; Wang, Y. Background subtraction based on adaptive non-parametric model. In Proceedings of the 7th World Congress on Intelligent Control and Automation, (WCICA’08), Chongqing, China, 25–27 June 2008; pp. 5960–5965. [CrossRef]

28. Otsu, N. Threshold Selection Method from Gray-Level Histograms. *IEEE Trans. Syst. Man Cybern.* 1979, SMC-9, 62–66. [CrossRef]

29. GRAM Road-Traffic Monitoring. Available online: http://agamenon.tsc.uah.es/Personales/rlopez/data/rtm/ (accessed on 24 January 2021).

30. Gonzalez, R.C.; Woods, R.E. Morphological Image Processing. In *Digital Image Processing*, 4th ed.; Pearson Prentice Hall: New York, NY, USA, 2007; pp. 657–661.

31. Roess, R.P.; Prassas, E.S.; McShane, W.R. Road User and Vehicle Characteristics. In *Traffic Engineering*; Prentice Hall: Upper Saddle River, New Jersey, 2011; pp. 18–24.

32. AASHTO. *A Policy on Geometric Design of Highways and Streets*, 5th ed.; AASHTO: Washington, DC, USA, 2004

33. Wang, Y.; Jodoin, P.M.; Porikli, F.; Konrad, J.; Bezeth, Y.; Ishwar, P. CDnet 2014: An Expanded Change Detection Benchmark Dataset. In Proceedings of the 2014 IEEE Conference on Computer Vision and Pattern Recognition Workshops, (CVPRW 2014), Columbus, OH, USA, 23–28 June 2014; pp. 393–400. [CrossRef]

34. Quesada, J.; Rodriguez, P. Automatic Vehicle Counting Method Based on Principal Component Pursuit Background Modeling. In *2016 IEEE International Conference on Image Processing (icip)*; IEEE: New York, NY, USA, 2016; pp. 3822–3826.

35. Change Detection Benchmark Web Site. Available online: http://jacarini.dinf.usherbrooke.ca/dataset2014/ (accessed on 24 January 2020).

36. Videos for Experiments. Available online: https://pan.baidu.com/s/1Kq7R-paqIMOFhD8o4nJaDA (accessed on 24 January 2020).

37. Abdelwahab, M.A. Fast approach for efficient vehicle counting. *Electron. Lett.* 2019, 55, 20–21. [CrossRef]

38. Shakeri, M.; Zhang, H. COROLA: A sequential solution to moving object detection using low-rank approximation. *Award Sel. Pap. ACCV 2014* 2016, 146, 27–39. [CrossRef]

39. Yang, H.; Qu, S. Real-time vehicle detection and counting in complex traffic scenes using background subtraction model with low-rank decomposition. *IET Intel. Transp. Syst.* 2018, 12, 75–85. [CrossRef]

40. Gomaa, A.; Abdelwahab, M.M.; Abo-Zahhad, M.; Minematsu, T.; Taniguchi, R.I. Robust Vehicle Detection and Counting Algorithm Employing a Convolution Neural Network and Optical Flow. *Sensors* 2019, 19, 4588. [CrossRef] [PubMed]

41. Rosas-Arias, L.; Portillo-Portillo, J.; Hernandez-Suarez, A.; Olivaes-Mercado, J.; Sanchez-Perez, G.; Toscano-Medina, K.; Perez-Meana, H.; Sandoval Orozco, A.L.; Garcia Villafl, L.J. Vehicle Counting in Video Sequences: An Incremental Subspace Learning Approach. *Sensors* 2019, 19, 2848. [CrossRef] [PubMed]

42. Bouvié, C.; Scharcanski, J.; Barcellos, P.; Escouto, F.L. Tracking and counting vehicles in traffic video sequences using particle filtering. In Proceedings of the 2013 IEEE International Instrumentation and Measurement Technology Conference (I2MTC), Minneapolis, MN, USA, 6–9 May 2013; pp. 812–815. [CrossRef]

43. Harikrishnan, P.M.; Thomas, A.; Nisha, J.S.; Gopi, V.P.; Palanisamy, P. Pixel matching search algorithm for counting moving vehicle in highway traffic videos. *Multimed. Tools Appl.* 2020. [CrossRef]