

## Article

# Heterogeneous Traffic Condition Dataset Collection for Creating Road Capacity Value

Surya Michrandi Nasution <sup>1,2</sup>, Emir Husni <sup>1,\*</sup>, Kuspriyanto Kuspriyanto <sup>1</sup> and Rahadian Yusuf <sup>1</sup><sup>1</sup> School of Electrical Engineering and Informatics, Institut Teknologi Bandung, Jl. Ganesa 10, Bandung 40132, Indonesia<sup>2</sup> School of Electrical Engineering, Telkom University, Jl. Telekomunikasi, Bandung 40257, Indonesia

\* Correspondence: ehusni@itb.ac.id

**Abstract:** Indonesia has the third highest number of motorcycles, which means the traffic flow in Indonesia is heterogeneous. Traffic flow can specify its condition, whether it is a free flow or very heavy traffic. Traffic condition is the most important criterion used to find the best route from an origin to a destination. This paper collects the traffic condition for several road segments which are calculated based on the degree of saturation by using two methods, namely, (1) by counting the number of vehicles using object detection in the public closed-circuit television (CCTV) stream, and (2) by requesting the traffic information (vehicle's speed) using TomTom. Both methods deliver the saturation degree and calculate the traffic condition for each road segment. Based on the experiments, the average error rate obtained by counting the number of vehicles on Pramuka–Cihapit and Trunojoyo was 0–2 cars, 2–3 motorcycles, and 0–1 for others. Meanwhile, the average error on Merdeka-Aceh Intersection reached 6 cars, 11 motorcycles, and 1 for other vehicles. The average speed calculation for the left side of the road is more accurate than the right side, and the average speed on the left side is less than 3.3 km/h. Meanwhile, on the right side, the differences between actual and calculated vehicle speeds are between 11.088 and 22.222 km/h. This high error rate is caused by (1) the low resolution of the public CCTV, (2) some obstacles interfering with the view of CCTV, (3) the misdetection of the type of vehicles, and by (4) the vehicles moving too fast. The collected dataset can be used in further studies to solve the congestion problem, especially in Indonesia.

**Citation:** Nasution, S.M.; Husni, E.; Kuspriyanto, K.; Yusuf, R. Heterogeneous Traffic Condition Dataset Collection for Creating Road Capacity Value. *Big Data Cogn. Comput.* **2023**, *7*, 40. <https://doi.org/10.3390/bdcc7010040>

Academic Editors: Muhammad Syafrudin, Ganjar Alfian, Norma Latif Fitriyani and Muhammad Anshari

Received: 2 February 2023  
Revised: 17 February 2023  
Accepted: 20 February 2023  
Published: 22 February 2023



**Copyright:** © 2023 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

**Keywords:** heterogeneous traffic flow; traffic dataset collection; object detection; CCTV; TomTom; vehicle movement

## 1. Introduction

Traffic condition is the most important criterion in the selection of a travel route. It is common that drivers try to avoid congestion when driving to their destination. Traffic conditions can be determined by counting the flow in some areas, specifically by counting the number of vehicles in a certain period.

When counting the number of vehicles, there must be a categorization for each type of vehicle. Especially in Indonesia, where the characteristics of traffic flow are formed by several kinds of vehicles. Traffic flow in Indonesia is heterogeneous: it is not only formed by cars, but also formed by motorcycles, buses, and trucks. Based on the Indonesian Statistics Bureau in 2018, motorcycles dominate the traffic flow in Indonesia by 77.5% compared with other types of vehicles [1]. Based on the heterogeneous traffic flow in Indonesia, the number of vehicles that pass the road must be separated based on their type.

This paper uses a dataset collection method to investigate heterogeneous traffic flow so that it can be used as a training dataset for the prediction of future traffic conditions. The heterogeneous traffic condition dataset is rarely found in the transportation sector, and this paper delivers the first heterogeneous traffic dataset collected in Indonesia.

The measurement of traffic conditions is achieved by calculating the degree of saturation, which is obtained from information such as road width, road slope, traffic light cycle, etc. By combining this information with the default saturation, its degree is defined. This measurement is performed for each road segment in the observed area. Later, the dataset is stored to predict the traffic flow of a specific road segment.

The traffic dataset collection in this paper is achieved by choosing one of these methods: (1) implementing object detection and tracking in public CCTV systems that observed an intersection or a road segment, or by (2) collecting travel speed from TomTom digital maps [2]. Later, the collected traffic information is processed to create the dataset. The first method is applied whenever the area is surveilled by CCTV. Meanwhile, the information collected using TomTom is obtained for each road segment that cannot be observed by the public CCTV systems. The observation area in this paper covered 265 road segments and 89 intersections which are located around Jl. R.E. Martadinata, Bandung, Indonesia. The traffic condition dataset is separated for each road segment; therefore, 265 datasets were collected using one of these two methods.

As aforementioned, the use of CCTV to count vehicles is performed by implementing the object detection and tracking method. This method is commonly implemented to understand the situation using a computer. Several methods can be used to study the situation, namely YOLOv3 [3], RetinaNet [4], etc. In our heterogeneous traffic condition dataset collection, YOLOv3 was the method used in object detection and tracking. YOLOv3 can be implemented in various detection, such as face detection [5,6], text detection [7,8], and traffic light and sign detection [9,10], etc. [11–13].

The output of this study is a dataset for each observed road segment. The main dataset consists of several road criteria, such as days, times, weather conditions, and traffic conditions. Other information gathered along with this dataset collection is the calculation of the vehicle's velocity. The traffic dataset is used as training data to predict its future condition using any machine learning method; meanwhile, the average vehicle speed will work as subsidiary information in calculating the whole road condition [14].

This paper is organized as follows. The literature review discusses research which is related to the study in Section 2. The proposed concepts of the collection of traffic conditions is discussed in Section 3, followed by a discussion of the experimental result in Section 4. Finally, in Section 5, a conclusion is presented along with the results of the dataset collection based on heterogeneous traffic conditions.

## 2. Literature Review

### 2.1. Degree of Saturation

#### 2.1.1. Traffic Flow

Traffic condition in Indonesia is regulated based on the Indonesian Highway Capacity Manual called “Manual Kapasitas Jalan Indonesia” (MKJI). In order to measure the level of traffic conditions, the degree of saturation ( $DS$ ) must be defined first as its reference value. The value of  $DS$  is the comparison result of traffic flow and the observed road capacity [15], as shown in Equation (1).

$$DS = \frac{Q}{C} \quad (1)$$

The value of  $Q$  describes the traffic flow which is commonly measured in an hourly time span. Meanwhile,  $C$  represents the observed road segment capacity. Since the traffic flow in Indonesia is categorized as heterogeneous, the variances of vehicles on the road must be converted by calculating their number with the equivalency number, as shown in Table 1 [16–18].

**Table 1.** Vehicle type equivalency number.

| Vehicle Type     | Equivalency |
|------------------|-------------|
| Car (LV)         | 1           |
| Motorcycle (MC)  | 0.2         |
| Bus & Truck (HV) | 1.3         |

By finding the equivalent number of each vehicle, the traffic flow is determined. In this step, the traffic flow is converted from heterogeneous to homogeneous. The result of this step could be used to define the specific capacity of an observed road. Overall, Equation (2) is used to convert the traffic flow.

$$Q = (1.0 \times LV) + (0.2 \times MC) + (1.3 \times HV) \quad (2)$$

### 2.1.2. Road Capacity

As mentioned, the traffic condition is measured based on a comparison between the hourly traffic flow and the capacity of the vehicle on a road segment. When calculating the capacity of a road segment, this could be achieved by measuring the saturated traffic flow and the cycle of traffic lights.

Before calculating the saturated traffic flow, the base saturation must be defined first. According to the MKJI, the basis of traffic flow in Indonesia can be determined by using Equation (3). However, based on the real traffic condition, this equation is no longer appropriate [19]. According to Munawar, the constant value in the equation must be rearranged as shown in Equation (4) [20] so that it can describe the real base saturation of the observed road.

$$S_o = 600 \times RoadWidth \quad (3)$$

$$S_o = 780 \times RoadWidth \quad (4)$$

The value of the base saturation is used to calculate the value of the saturated traffic flow on an observed road segment. It is determined by collating several factors that affect the traffic flow, such as city size, side fraction, road slope, right and left turn, etc. [21]. Equation (5) is used to measure the saturated traffic flow based on the base saturation value and other factors [22,23].

$$S = S_o \times F_{CS} \times F_{SF} \times F_G \times F_P \times F_{RT} \times F_{LT} \quad (5)$$

Table 2 shows an example of defining a city size factor. Its value is determined based on the number of citizens in the observed city. As seen in the table, its value will become greater if there is an increment in the number of citizens.

**Table 2.** City size factor for saturated flow calculation.

| Number of Citizens (Millions) | City Size Factor |
|-------------------------------|------------------|
| >3.0                          | 1.05             |
| 1.0–3.0                       | 1                |
| 0.5–1.0                       | 0.94             |
| 0.1–0.5                       | 0.83             |
| <0.1                          | 0.82             |

Other factors also directly affect the saturated traffic flow; for example, the side fraction can make a distraction for the vehicles passing on the road. The parking area which commonly appears on the road also has an effect on the traffic flow. Generally, each factor has a direct impact on the calculation of the saturated traffic flow.

The other parameter that can be used to determine road capacity is the traffic light cycle. It commonly appears in intersections, which is also the best location for observing the road when using public CCTV systems. The location of public CCTV in Bandung is dominated by intersections, yet there are also several CCTV systems that observe road segment directly.

While there is an impact on the calculation depending on the road capacity for each location, the CCTV systems at the intersection have a greater impact. Its cycles could determine the road capacity along with the saturated traffic flow that was previously explained. As shown in Equation (6), the capacity of the road is affected by the saturated traffic flow, the time of green light ( $g$ ), and the traffic light cycle ( $c$ ) (red-to-red light cycle time). Meanwhile, whenever the CCTV systems are not located in an intersection, their cycles can be defined as 1.

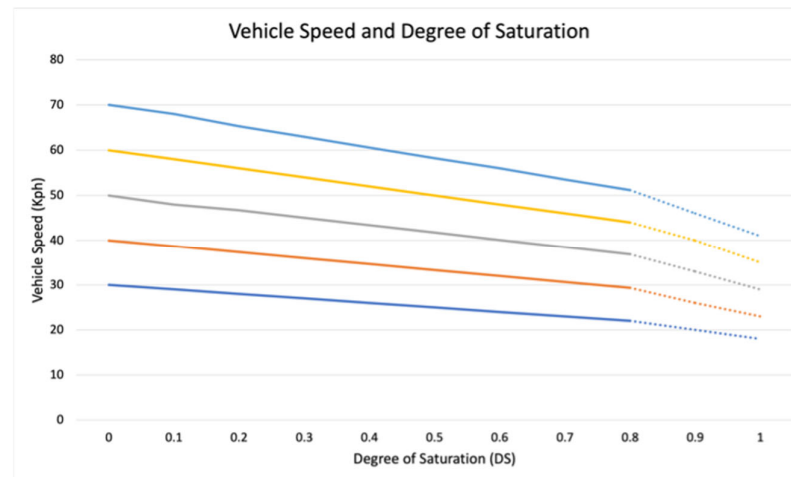
$$C = S \times \frac{g}{c} \quad (6)$$

Equation (7) shows the determination of the saturation degree, which is obtained by substituting Equation (1) with Equation (6). This equation could be used to define the saturation degree, either in an intersection or in the middle of the road.

$$DS = \frac{Q \times c}{S \times g} \quad (7)$$

### 2.1.3. Degree of Saturation Based on Road Infrastructure

The previous method is only applied when the value of saturation and other factors are clearly defined, such as the numbers of each vehicle, road width, etc. On the other hand, the degree of saturation also can be calculated by using the rule from the MKJI. As shown in Figure 1, the relationship between vehicle speed and the saturation degree is defined. This relationship only applies to the undivided two-lane and two-way roads (2/2UD) [16].



**Figure 1.** The rule of vehicle speed and saturation degree.

According to Nasution et al., the relationship between the velocity of a vehicle and its density is inversed [24]. This means that, whenever the velocity starts to decline, the value of density increases immediately. As mentioned before, regarding the two-lane and two-way roads, the relationship between these two parameters is linear (solid-colored lines) until the density reached 80%. Its relationship changes when the density is greater than 80% (dotted lines). Based on Figure 1, this relationship can be formulated as a line equation for various velocities (30, 40, 50, 60, and 70 km/h). Table 3 shows the equation

for each speed of the vehicle, which is shown in Figure 1. It is only valid for density that is less than 80%.

**Table 3.** Line equation according to vehicle speed and saturation degree.

| Vehicle Speed (KpH) | Line Equation       |
|---------------------|---------------------|
| 30                  | $y = -10x + 30$     |
| 40                  | $y = -13.33x + 40$  |
| 50                  | $y = -16.471x + 50$ |
| 60                  | $y = -20x + 60$     |
| 70                  | $y = -23.529x + 70$ |

According to several line equations shown in Table 3, there is a general equation that can be used for any vehicle's speed, which is shown in Equation (8).  $S_f$  is the free flow speed on the roads,  $y$  is the current speed detected on the road, and  $x$  is the value of density in a specific vehicle's speed.

$$y = \left(-\frac{1}{3}S_fx\right) + S_f \quad (8)$$

If  $y$  is equal to current speed ( $S$ ) and  $x$  is equal to density ( $DS$ ), the previous equation is changed, as shown in Equation (9). Meanwhile, by converting this equation, the level of density can be determined using Equation (10) [25,26].

$$S = \left(-\frac{1}{3}S_fDS\right) + S_f \quad (9)$$

$$DS = 3\left(1 - \frac{S}{S_f}\right) \quad (10)$$

There is a categorization based on the value of density in order to define the traffic condition. In general, there are six categories ( $A$  to  $F$ ) used to define its condition. In this paper, we divided the traffic condition into four categories, namely, "Freeflow" (0), "Medium" (1), "Heavy" (2), and "Very Heavy" (3). This simplification in the category is performed to reduce the data distribution so that it can later enhance the performance when it is used as a dataset in a traffic prediction system. The categorization of traffic conditions ( $TC$ ) is defined using Equation (11).

$$TC = \begin{cases} \text{Freeflow Traffic (0),} & DS < 0.25 \\ \text{Medium Traffic (1),} & 0.25 \leq DS < 0.5 \\ \text{Heavy Traffic (2),} & 0.5 \leq DS < 0.75 \\ \text{Very Heavy Traffic (3),} & DS \geq 0.75 \end{cases} \quad (11)$$

## 2.2. Video Analysis with Object Detection

Video analysis is a computer vision method that allows the computer to calculate the situation. In our analysis, a video was commonly divided into several frames. For each frame, there was a process to detect the objects which was able to classify the objects that appeared in the frame. There were 91 common objects stored in a dataset by Microsoft, called Common Objects in Context (COCO) [27].

Object detection is commonly applied in order to recognize objects using a camera. This method can detect pedestrians on the street [28–30], recognize faces to prevent burglary or access to a special room [31,32], detect texts [33], etc. [34]. There are lots of methods that can be applied to detect objects, namely the Histogram of Oriented Gradients (HOGs) [35], RetinaNet [36], Region-based Convolutional Neural Networks (R-CNNs) [37], Single Shot Detection (SSD) [38], YOLO [39,40], etc.

Of these methods, the most famous one was proposed by Redmon, called YOLO [3]. This is a very popular method because it can detect objects faster than other YOLO

versions [41]. On the other hand, it still has almost similar performances (average precision) compared to the others [42]. Several researchers have also tried to modify this method in order to adapt to small devices such as Raspberry or Jetson [32,43,44].

This paper implements YOLOv3 to detect objects. The previous version of this method (YOLOv1) has two layers that are fully connected and 24 convolutional layers [3]. Meanwhile, in YOLOv2, the fully connected layers are removed. In addition, the anchor boxes which are used to predict the object's bounding boxes are introduced [45]. In YOLOv3, there is a residual structure that is used to enhance the depth of the network layer [39]. Based on this enhancement, YOLOv3 achieves a breakthrough in its performance. YOLOv3 is the last version of Redmon's original algorithm; meanwhile, the newer version is not developed by Redmon. The newer YOLO version has various goals, according to the developers.

The implementation of object detection in transportation areas (especially on a road-way) can be applied by using a public or private camera. A public camera refers to any CCTV systems that monitor the situation of an area. By using this type of camera, macroscopic traffic information can be collected [46,47]. Meanwhile, the private one refers to any capturing device that is applied only for a specified driver, such as an in-car dashcam, smartphone, etc. Since it refers to personal information, it delivers microscopic information [48–50].

Both cameras collect a large amount of images or video. Based on its collection, and after the preprocessing and feature extraction had begun, the objects that appeared on the image or video could be recognized. The output of this process can be used as an input in the next step, such as for vehicle counting, to estimate the crowd [51–54], to detect a vehicle's license number to prevent traffic violations [55,56], for traffic light detection for managing the traffic [57], and for road-type detection for driving convenience [58], etc.

### 3. Proposed Framework

The dataset was collected to predict the current traffic condition. The framework proposed in this paper collected several road criteria which determined the road situation, such as days, rush hour, weather, and the density itself. Figure 2 shows the process of the dataset collection framework proposed in this paper.

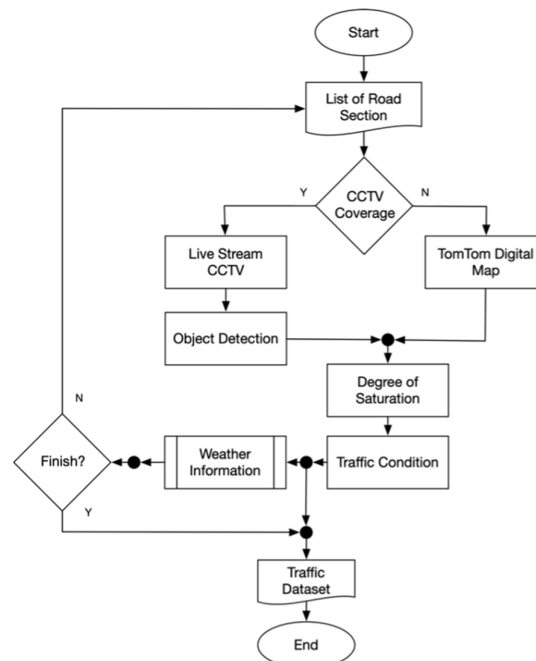


Figure 2. The proposed traffic dataset collection framework.

In the beginning, the system assessed the CCTV's coverage of the road sections that are listed. When an area was covered by CCTV, object detection and vehicle tracking were performed on the CCTV's stream to calculate the traffic condition. On the other hand, the traffic condition on the road which was not covered by CCTV was collected and determined using TomTom Digital Map. Whenever the saturation degree and traffic condition were defined, the system collected the current weather condition based on OpenWeather. This section describes each collection method that was used in this paper.

### 3.1. Dataset Collection Using Object Detection in CCTV

The traffic condition dataset was obtained by examining the road situation using a computer vision method, namely, object detection and object tracking. Based on the video or image from CCTV, a computer can recognize the objects that appear in the image frame. Each CCTV location can deliver its condition by using different compositions since the angle of the CCTV will be placed in a different position. As shown in Figure 3, there is a sample image from a CCTV on the observed road. Two vehicles pass an intersection; meanwhile, there is a queue on the road segments, and there are also several vehicles that are parked on the lot.



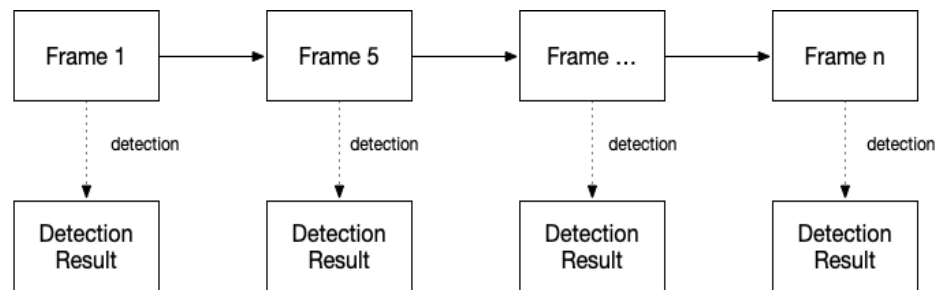
**Figure 3.** Sample Frame of CCTV Stream.

CCTV streams are opened and read by the OpenCV module, and every detected object in each road segment is counted and stored in the database. This module allows the system to calculate the objects which are limited to the common object in context (COCO) [27]. The objects the system attempted to recognize are restricted to several types of vehicles, such as cars, motorcycles, trucks, and buses.

#### 3.1.1. Vehicle Counting

The process of detecting objects based on the CCTV's streams was conducted in 5 s with 25 frames per second (fps) as its framerate. The total frame stored in the database is  $\pm 125$  frames. To save computer resources (CPU, RAM, etc.) and computational time, object detection was performed every five frames.

Figure 4 shows an illustration of the detection process that occurred in the proposed framework. Vehicle detection determines its numbers on the road every five frames. By using this concept, an average number of vehicles in 5 s of CCTV streams is obtained. Every object detected in the system was allocated a bounding box that shows the detected object boundary.



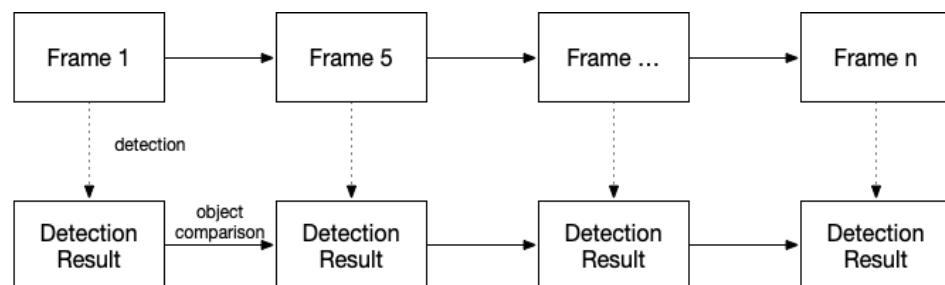
**Figure 4.** The process of vehicle counting.

Apparently, without any further modification of the detecting objects, errors can occur in counting of the vehicles. This is because a vehicle that parks in the CCTV coverage zone is counted as a vehicle that passed on the observed road, as shown in Figure 5. Therefore, a method that excludes parking vehicles is required to obtain the real traffic flow.



**Figure 5.** Detection result on the frame of CCTV stream.

In response to this problem, we made a slight modification in the object detection method. We modified the previous method by comparing objects between two adjacent frames. Figure 6 shows the object comparison between two frames (e.g., frame 1 and frame 5). If there is a movement between these frames, the number of vehicles is added by its detection result. However, if the position of objects is almost similar between the two frames, it will be excluded from the calculation.

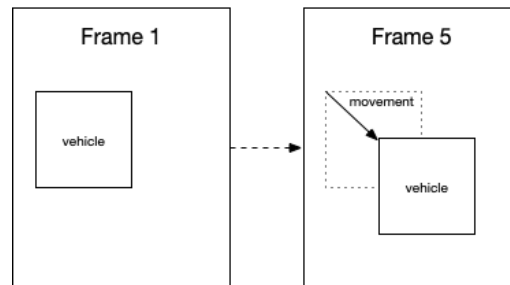


**Figure 6.** Object comparison between two adjacent frames.

The comparison between the two frames was conducted from the very beginning of the process until it is finished. By using this concept, the position of detected vehicles in

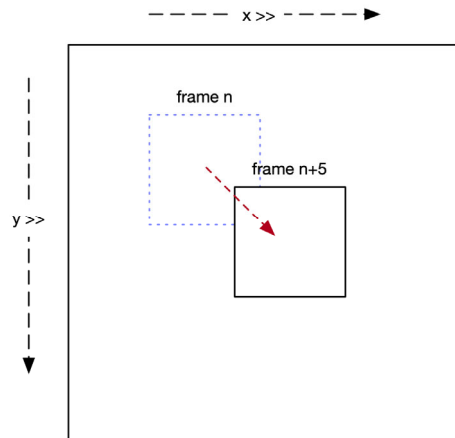
the frame  $n + 5$  could be compared with the next frame ( $n + 10$ ). This also applies for vehicles that are newly detected in the middle of the detection process. In the end, it delivers the number of vehicles that pass the observed road.

We divided the vehicle's direction by tracking its movement. Movement detection also uses a similar concept to the previous method. By comparing two adjacent frames, its direction is obtained. Figure 7 illustrates the tracking of the movement of a vehicle. A vehicle is considered a moving object if it has moved between its thresholds. If an object's movement is under the threshold, the vehicle is considered to be stopped. On the other hand, if the distance is above its threshold, it is considered a new vehicle that is barely recognized by the system.



**Figure 7.** Vehicle's movement detection.

The position of the detected vehicle is shown in cartesian, which has the value of  $x$  and  $y$ . As seen in Figure 8, in frame  $n + 5$ , the position has a greater value of  $x$  and  $y$  than the previous frame ( $n$ ).



**Figure 8.** The vehicle movement measurement.

As aforementioned, the movement of a vehicle is tracked based on the centroid of the detected object boundary (object's bounding box, as shown in Figure 5). Its movement is defined based on the position in the next frame. To understand a vehicle's movements, the slope (gradient) between the centroids is calculated using Equation (12).

$$m = \frac{y_2 - y_1}{x_2 - x_1} \quad (12)$$

The result of the gradient measurement delivers different vehicle movements as shown in Table 4. There are several schemes of CCTV direction, such as those that are perpendicular, parallel, and diagonal (left and right) with the road. The gradient value determines which lane a vehicle passed.

For CCTV which has a parallel position with the road, its gradient value is 0 ( $m = 0$ ), therefore the vehicle movement is defined based on the change of  $x$ . Whenever it has a positive value (+), it moves from the left side to the right side of the CCTV observation area, and vice versa. Meanwhile, when the position of CCTV is perpendicular to the road, the value of the gradient is set as  $\infty$  ( $m = \infty$ ). The movement of a vehicle is defined by the change of value  $y$ .

In reality, CCTV positions perpendicular and parallel are rarely found. Most CCTV systems are placed at the edge of an intersection. The vehicles move diagonally either left or right. Based on this situation, the gradient value is either more than 0 or less than 0. If the gradient is greater than 0 ( $m > 0$ ), it means that the increment in both the  $x$  and  $y$  values illustrates the movement from the left side to the right side of the CCTV's point of view. On the other hand, when the gradient is less than 0 ( $m < 0$ ), the value of  $x$  and  $y$  cannot increase at the same time. If the  $x$  is increased, then the value of  $y$  must be decreased. Overall, in this kind of CCTV position, the side of the road is determined, as shown in Table 4.

**Table 4.** The calculation of vehicle direction based on movement gradient.

| Gradient Value | CCTV Point of View                                                                  | $\Delta X$ | $\Delta Y$ | Vehicle Movement Direction |
|----------------|-------------------------------------------------------------------------------------|------------|------------|----------------------------|
| $m = \infty$   |    | 0          | +          | Right side                 |
|                |                                                                                     | 0          | −          | Left side                  |
| $m = 0$        |    | +          | 0          | Right side                 |
|                |                                                                                     | −          | 0          | Left side                  |
|                |                                                                                     | −          | −          | Left side                  |
| $m > 0$        |   | +          | +          | Right side                 |
|                |                                                                                     | −          | −          | Left side                  |
|                |                                                                                     | +          | +          | Right side                 |
|                |                                                                                     | −          | +          | Right side                 |
|                |                                                                                     | +          | −          | Left side                  |
| $m < 0$        |  | −          | +          | Right side                 |
|                |                                                                                     | +          | −          | Left side                  |

By separating the movements, the number of vehicles can be determined for each roadside. This number is used to define the saturation degree and the traffic condition. Later, the methods of vehicle movement tracking can also be used to calculate the speed of the vehicle.

### 3.1.2. The Calculation of Vehicle Speed

As mentioned in the previous section, the vehicle speed calculation method is similar to the vehicle tracking movement method, which compared the centroid between two adjacent frames. At first, the distance (pixels) is measured using Euclidian Distance by using the coordinates between two frames,  $(x_1, y_1)$  and  $(x_2, y_2)$ . The  $x_1$  and  $y_1$  values of an object come from the centroid in the first frame; meanwhile,  $x_2$  and  $y_2$  are the coordinates of the object in the next frame.

$$Distance(px) = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2} \quad (13)$$

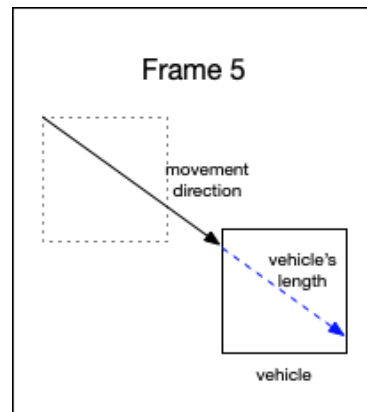
Equation (13) is used to determine the movement distance of an object between two frames. There must be an adjustment in the distance value since the outcome of the

equation is in pixels. Moreover, there must be a conversion from pixels into meters. Table 5 shows the common vehicle length that appears in Indonesia.

**Table 5.** Common vehicle length in Indonesia.

| Vehicle Type | Vehicle Length (Meters) |
|--------------|-------------------------|
| Car          | 4.5                     |
| Motorcycle   | 2.2                     |
| Bus          | 12.5                    |
| Truck        | 12.19                   |

To convert the movement distance (from pixels to meters), the vehicle length (in pixels and meters) and its movement distance (in pixels) are required. The length of the vehicle (in pixels) is measured based on its bounding box size and is calculated according to the vehicle's movement direction, as illustrated in Figure 9. The blue-dashed line in the figure is the length of the vehicle, which is obtained from its movement projection. When these values are gathered, the conversion of the distance is began using Equation (14).



**Figure 9.** The vehicle speed measurement.

$$Distance(m) = \frac{Vehicle\ Length\ (m)}{Vehicle\ Length\ (px)} \times Distance\ (px) \quad (14)$$

The vehicle speed ( $m/s$ ) is defined based on its movement distance ( $m$ ), the framerate of the CCTV, and the timespan ( $s$ ). In this paper, the stream duration is around 5 s and consists of 25 frames for each second. Based on this situation, five frames represent the road situation in each second, which means that every two adjacent frames represent 0.2 s for its period. Equation (15) is used to calculate the vehicle speed in the observed area using public CCTV.

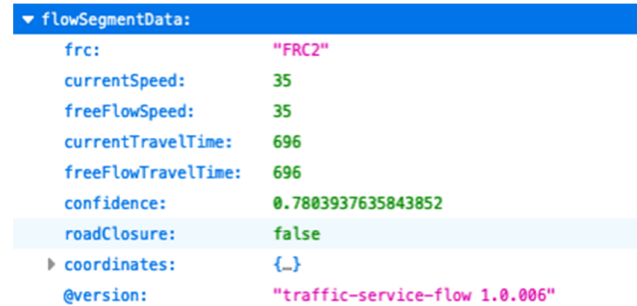
$$VehicleSpeed\ (m/s) = \frac{Distance\ (m)}{Time\ (s)} \times \frac{FPS}{frame\ sampling} \quad (15)$$

The speed calculation is performed for every detected vehicle that appeared in two adjacent frames. In the end, there are several average speeds calculated from each vehicle type detected during the measurement.

### 3.2. Dataset Collection Using TomTom Digital Maps

The TomTom digital maps application is commonly used on smartphones (iPhone) or in-vehicle navigation systems. There are several road criteria provided by TomTom, such as vehicle speed and time travel for current conditions and when there is no vehicle on the road (*freeflow*), etc.

These criteria can be collected by using the coordinates (latitude and longitude) of the road. The proposed framework sends a request to TomTom's server, and obtains responses related to the road criteria, as mentioned above. Figure 10 shows the response from TomTom's server.



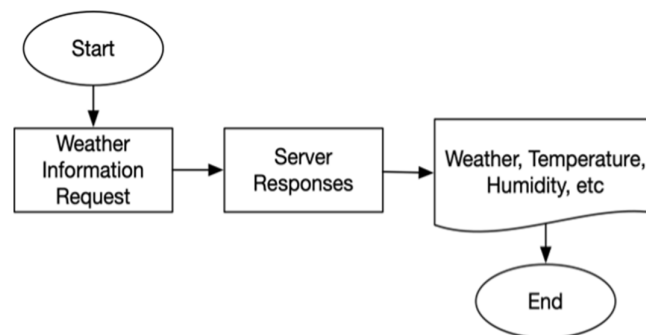
**Figure 10.** The response from TomTom Digital Maps server.

It can be seen from the figure that there is a road with criteria as follows: current speed: 35 m/s, free flow speed: 35 m/s, current travel time: 696 s, free flow travel time: 696 s, confidence: 78%, road closure: false, and its coordinates.

### 3.3. Weather Information

Weather is another important road criterion that must be considered in creating a dataset. For example, high rainfall can restrain the distance of visibility and can also make the velocity of light and heavy vehicles decline. On the other hand, rain makes lots of motorcycle drivers choose to stop driving because the road becomes slippery. Because of this situation, sometimes, there are no motorcycles on the road, leaving the road empty.

The weather information was collected using OpenWeather, which delivers its real-time condition. The process of gathering weather information is similar when collecting information on the road using TomTom digital maps, which is illustrated in Figure 11. There is a request for a specific location based on its coordinates.

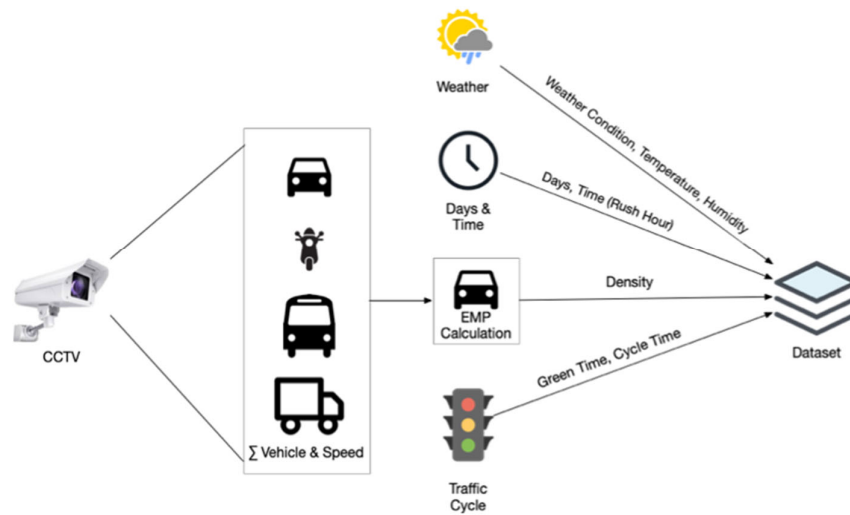


**Figure 11.** The process of weather information collection.

### 3.4. The Compilation of Traffic Dataset

The collected data are processed to determine the saturation degree and the category of traffic conditions. The criteria collected in this paper are days, rush hour, weather conditions, the number of vehicles, vehicle speed, and traffic conditions.

As mentioned in the previous section, the process of data compilation using CCTV and TomTom is different. Traffic condition based on CCTV is calculated based on the number of vehicles that pass during the observation time. Meanwhile, the other is calculated based on the line equation according to the Indonesian Highway Capacity Manual (MKJI). Figure 12 shows the process of dataset compilation using public CCTV.



**Figure 12.** Traffic Dataset Compilation using Public CCTVs.

In this study, the observation area is in Bandung, Indonesia. According to the Indonesian Bureau of Statistics, in 2019, the number of citizens in Bandung was 2.5 million [59]. This value meant that the city size factor in the saturated degree was set as 1. The other factor that is defined in calculating the saturated traffic flow is the side fraction, which was set at 0.94. This indicates that the observed roads are categorized as medium roads and that they also have a lower ratio of nonmotorized vehicles on the roads. Meanwhile, the remaining factors are set to 1; moreover, since the road is categorized as a non-inclined road, it is also forbidden to park any vehicle on the side of the road, and we assumed that there was no interruption in the left or right turn in the intersection.

Based on these considerations, Equation (5) was reconstructed using the value of the road factors that were previously defined. Equation (16) determined the saturated degree on each road segment. Each road segment has a different traffic flow saturation ( $S_o$ ) based on its width.

$$S = S_o \times 0.94 \quad (16)$$

Before determining the level of traffic conditions, the degree of saturation must be defined. Along with the value of saturated traffic flow, the number of vehicles, traffic light cycles, and green time are used to determine the saturation degree. The number of vehicles used in this calculation must be processed to generalize its number (from heterogeneous to homogeneous). The time of traffic light cycles and green lights is manually observed. By calculating these values using Equation (7), the measurement of saturation degree using public CCTV is determined.

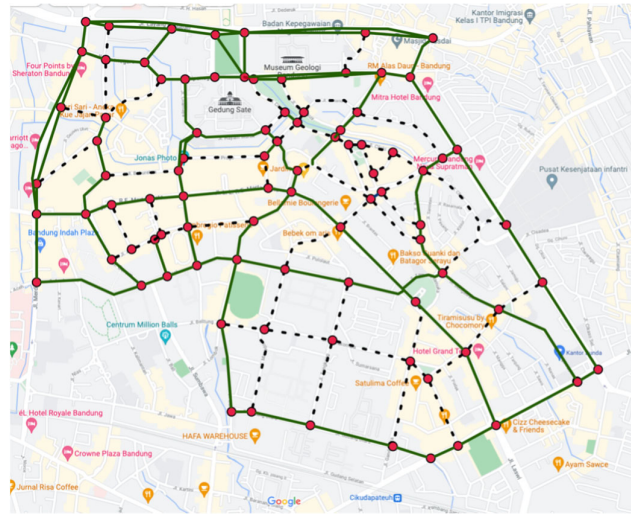
On the other hand, the measurement of the saturation degree using TomTom digital maps is simpler than the calculation of the saturation degree using public CCTV, as it only requires the free flow and current speed on a road segment. By inserting these values into Equation (10), the degree of saturation using TomTom digital maps is specified.

Both saturation degree calculations are used to define the category of traffic conditions. Later, along with other criteria, the level of traffic condition is stored in the traffic dataset for each road segment.

## 4. Experimental Results

### 4.1. Traffic Condition Dataset Collection

As previously mentioned at the beginning of this paper, the dataset collection is conducted on several road segments in Jl. R.E. Martadinata, Bandung, Indonesia. As shown in Figure 13, the observation area includes 89 intersections (red circles) that connect 265 road segments. The green-line roads can accommodate every kind of vehicle, while the roads with the black-dashed lines are specified for motorcycles.



**Figure 13.** Observation area.

Table 6 shows the roads covered by public CCTV. Whenever there is CCTV coverage on the roads, the level of traffic condition is measured based on its streams. Meanwhile, for the other roads where there is no CCTV coverage, traffic condition is calculated using TomTom digital maps.

**Table 6.** Several covered roads using CCTV.

| No | Observed Area             | Type CCTV | No | Observed Area                | Type CCTV |
|----|---------------------------|-----------|----|------------------------------|-----------|
| 1  | Merdeka-Aceh Intersection | PTZ       | 6  | Cihapit Intersection         | PTZ       |
| 2  | Aceh Intersection         | PTZ       | 7  | Angrek Road Segment          | Static    |
| 3  | Trunojoyo Intersection    | PTZ       | 8  | Pramuka–Cihapit Road Segment | Static    |
| 4  | Trunojoyo Road Segment    | PTZ       | 9  | Telkom Intersection          | Static    |
| 5  | Lombok Intersection       | PTZ       | 10 | Banda Intersection           | Static    |

Regarding the roads that were covered by CCTV, a validation was obtained for the selected intersections. We tried to compare the calculation of traffic conditions using CCTV with their real condition. The Pramuka–Cihapit and Trunojoyo road segments and The Merdeka-Aceh intersection were chosen to validate our proposed framework. Pramuka–Cihapit is the chosen road segment which is located on the main road of Jl. R.E. Martadinata. There are no traffic lights in this area, therefore the green time and traffic light cycle are set as 1. The CCTV in Trunojoyo is placed near an intersection that has a famous supermarket and hospital, therefore this area is categorized as a crowded area. At the last CCTV location, the Merdeka-Aceh intersection is a one-way road. The CCTV in this intersection is a Pan-Tilt-Zoom (PTZ) CCTV, therefore the operator of this kind of CCTV is able to change its point of view. Near this intersection, there is an attractive place (a mall); however, since it has a one-way route, the mall does not affect the traffic condition.

As a validation method, we calculated the performance of the vehicle counting system and its speed by comparing it to the actual condition. The actual condition was acquired through a manual observation of the number of vehicles and their speed. The mean absolute error (MAE) was the method used to validate our results. This validation method was only applied for observations based on public CCTV.

#### 4.1.1. Validation of Traffic Condition Based on CCTV

The validation was conducted at least 30 times for every selected CCTV system. As mentioned before, this validation calculates the error using the mean absolute error (MAE) between the proposed framework and the actual result. The MAE is measured using Equation (17).  $g'_i$  is the result from our framework and  $g_i$  is the actual condition that appeared in the CCTV. In the end, the average absolute error rate was determined [60].

$$MAE(g, g') = \frac{1}{k} \sum_{j=1}^k |g_i - g'_i| \quad (17)$$

#### Pramuka–Cihapit Road Segment

As mentioned in the previous section, this CCTV system was placed in a road segment, which makes the point of the CCTV system's view always parallel with the lane. This road is divided into two lanes, namely, vertical left (VL), where vehicles are leaving the CCTV, and vertical right (VR), which has a similar meaning to the movement of a vehicle coming to the CCTV. To determine the performance in this road segment, we compared the result 30 times at different times. Table 7 shows a comparison of the number of vehicles between our proposed framework and its actual value.

**Table 7.** Comparison between Detection and Actual Result on Pramuka–Cihapit Road Segment.

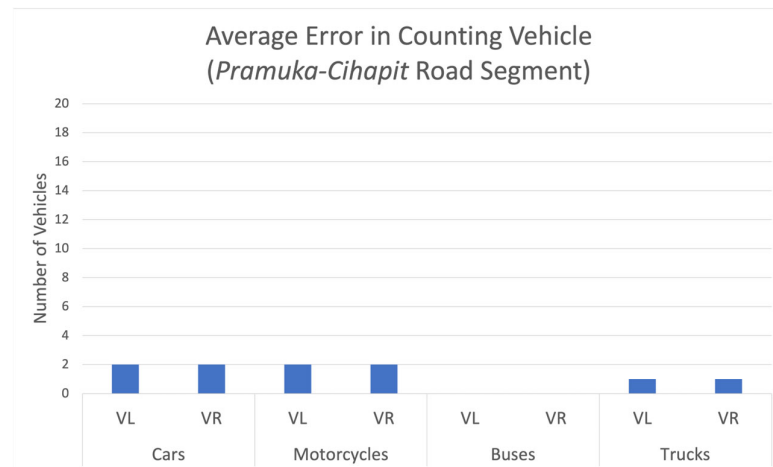
| Detection |    |            |    |     |    |       |    | Actual |    |            |    |     |    |       |    |
|-----------|----|------------|----|-----|----|-------|----|--------|----|------------|----|-----|----|-------|----|
| Car       |    | Motorcycle |    | Bus |    | Truck |    | Car    |    | Motorcycle |    | Bus |    | Truck |    |
| VL        | VR | VL         | VR | VL  | VR | VL    | VR | VL     | VR | VL         | VR | VL  | VR | VL    | VR |
| 2         | 2  | 4          | 1  | 0   | 0  | 0     | 0  | 2      | 6  | 7          | 6  | 0   | 0  | 0     | 0  |
| 1         | 1  | 0          | 1  | 0   | 0  | 0     | 0  | 1      | 2  | 2          | 2  | 0   | 0  | 0     | 0  |
| 5         | 0  | 2          | 1  | 1   | 0  | 0     | 0  | 9      | 0  | 1          | 7  | 0   | 0  | 0     | 0  |
| 4         | 0  | 2          | 0  | 0   | 0  | 0     | 0  | 5      | 3  | 8          | 1  | 0   | 0  | 0     | 0  |
| 1         | 3  | 2          | 2  | 0   | 1  | 0     | 0  | 2      | 4  | 3          | 11 | 1   | 0  | 0     | 0  |
| 2         | 5  | 0          | 1  | 0   | 0  | 0     | 0  | 4      | 7  | 1          | 2  | 0   | 0  | 0     | 0  |
| 2         | 1  | 0          | 0  | 0   | 0  | 0     | 0  | 4      | 1  | 0          | 0  | 0   | 0  | 0     | 0  |
| 2         | 1  | 0          | 1  | 1   | 0  | 0     | 0  | 3      | 4  | 0          | 5  | 1   | 0  | 0     | 0  |
| 5         | 0  | 1          | 1  | 0   | 0  | 0     | 0  | 6      | 2  | 3          | 2  | 0   | 0  | 0     | 0  |
| 3         | 2  | 1          | 1  | 0   | 0  | 0     | 0  | 5      | 7  | 1          | 2  | 0   | 0  | 0     | 0  |
| 4         | 0  | 1          | 4  | 0   | 0  | 0     | 0  | 7      | 0  | 4          | 8  | 0   | 0  | 0     | 0  |
| 5         | 5  | 2          | 2  | 0   | 0  | 0     | 0  | 6      | 8  | 2          | 8  | 0   | 0  | 0     | 0  |
| 5         | 2  | 5          | 1  | 0   | 0  | 0     | 0  | 3      | 2  | 10         | 1  | 0   | 0  | 0     | 0  |
| 3         | 0  | 3          | 0  | 0   | 0  | 0     | 0  | 3      | 1  | 4          | 0  | 0   | 0  | 0     | 0  |
| 0         | 2  | 2          | 1  | 0   | 0  | 0     | 0  | 0      | 6  | 2          | 6  | 0   | 0  | 0     | 0  |
| 1         | 2  | 1          | 2  | 0   | 0  | 0     | 0  | 2      | 4  | 3          | 4  | 0   | 0  | 0     | 0  |
| 2         | 1  | 0          | 0  | 0   | 0  | 0     | 0  | 4      | 1  | 1          | 6  | 0   | 0  | 0     | 0  |
| 1         | 2  | 0          | 0  | 0   | 0  | 0     | 0  | 3      | 4  | 1          | 2  | 0   | 0  | 0     | 0  |
| 1         | 0  | 1          | 0  | 0   | 0  | 0     | 0  | 2      | 1  | 5          | 0  | 0   | 0  | 0     | 0  |
| 7         | 0  | 0          | 0  | 0   | 0  | 0     | 0  | 8      | 0  | 0          | 0  | 0   | 0  | 0     | 0  |
| 3         | 1  | 6          | 1  | 1   | 0  | 0     | 0  | 0      | 3  | 1          | 7  | 1   | 0  | 0     | 0  |
| 5         | 0  | 0          | 0  | 0   | 0  | 0     | 0  | 5      | 3  | 0          | 1  | 0   | 0  | 0     | 0  |

|   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 3 | 0 | 0 | 2 | 0 | 0 | 0 | 0 | 3 | 0 | 0 | 2 | 0 | 0 | 0 | 0 |
| 3 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 4 | 2 | 2 | 1 | 0 | 0 | 0 | 0 |
| 4 | 3 | 0 | 0 | 2 | 0 | 0 | 0 | 7 | 5 | 3 | 0 | 1 | 0 | 0 | 0 |
| 5 | 3 | 2 | 0 | 0 | 0 | 0 | 0 | 8 | 2 | 4 | 0 | 0 | 0 | 0 | 0 |
| 5 | 1 | 2 | 1 | 0 | 0 | 0 | 0 | 8 | 0 | 4 | 1 | 0 | 0 | 0 | 0 |
| 2 | 0 | 2 | 0 | 0 | 0 | 0 | 0 | 2 | 1 | 4 | 0 | 0 | 0 | 0 | 0 |
| 4 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 5 | 2 | 3 | 3 | 0 | 0 | 0 | 0 |
| 4 | 2 | 2 | 2 | 0 | 0 | 0 | 0 | 4 | 3 | 7 | 9 | 0 | 0 | 0 | 0 |

According to the experimental result, the calculation of the error rate was performed by separating the number of vehicles based on the lanes and their type. Table 8 shows the value of error based on the lanes and the vehicle type. Meanwhile, Figure 14 shows the average value of error in calculating the vehicles on this road segment.

**Table 8.** Error rate in vehicle counting on Pramuka–Cihapit road segment.

| Testing | Car |    | Motorcycle |    | Bus |    | Truck |    |
|---------|-----|----|------------|----|-----|----|-------|----|
|         | VL  | VR | VL         | VR | VL  | VR | VL    | VR |
| 1       | 0   | 4  | 3          | 5  | 0   | 0  | 0     | 0  |
| 2       | 0   | 1  | 2          | 1  | 0   | 0  | 0     | 0  |
| 3       | 4   | 0  | 1          | 6  | 0   | 0  | 1     | 0  |
| 4       | 1   | 3  | 6          | 1  | 0   | 0  | 0     | 0  |
| 5       | 1   | 1  | 1          | 9  | 0   | 0  | 1     | 1  |
| 6       | 2   | 2  | 1          | 1  | 0   | 0  | 0     | 0  |
| 7       | 2   | 0  | 0          | 0  | 0   | 0  | 0     | 0  |
| 8       | 1   | 3  | 0          | 4  | 0   | 0  | 0     | 0  |
| 9       | 1   | 2  | 2          | 1  | 0   | 0  | 0     | 0  |
| 10      | 2   | 5  | 0          | 1  | 0   | 0  | 0     | 0  |
| 11      | 3   | 0  | 3          | 4  | 0   | 0  | 0     | 0  |
| 12      | 1   | 3  | 0          | 6  | 0   | 0  | 0     | 0  |
| 13      | 2   | 0  | 5          | 0  | 0   | 0  | 0     | 0  |
| 14      | 0   | 1  | 1          | 0  | 0   | 0  | 0     | 0  |
| 15      | 0   | 4  | 0          | 5  | 0   | 0  | 0     | 0  |
| 16      | 1   | 2  | 2          | 2  | 0   | 0  | 0     | 0  |
| 17      | 2   | 0  | 1          | 6  | 0   | 0  | 0     | 0  |
| 18      | 2   | 2  | 1          | 2  | 0   | 0  | 0     | 0  |
| 19      | 1   | 1  | 4          | 0  | 0   | 0  | 0     | 0  |
| 20      | 1   | 0  | 0          | 0  | 0   | 0  | 0     | 0  |
| 21      | 3   | 2  | 1          | 0  | 0   | 0  | 0     | 0  |
| 22      | 0   | 3  | 0          | 1  | 0   | 0  | 0     | 0  |
| 23      | 0   | 0  | 0          | 0  | 0   | 0  | 0     | 0  |
| 24      | 1   | 1  | 1          | 1  | 0   | 0  | 0     | 0  |
| 25      | 3   | 2  | 3          | 0  | 0   | 0  | 1     | 0  |
| 26      | 3   | 1  | 2          | 0  | 0   | 0  | 0     | 0  |
| 27      | 3   | 1  | 2          | 0  | 0   | 0  | 0     | 0  |
| 28      | 0   | 1  | 2          | 0  | 0   | 0  | 0     | 0  |
| 29      | 1   | 2  | 3          | 2  | 0   | 0  | 0     | 0  |
| 30      | 0   | 1  | 5          | 7  | 0   | 0  | 0     | 0  |
| Average | 2   | 2  | 2          | 2  | 0   | 0  | 1     | 1  |



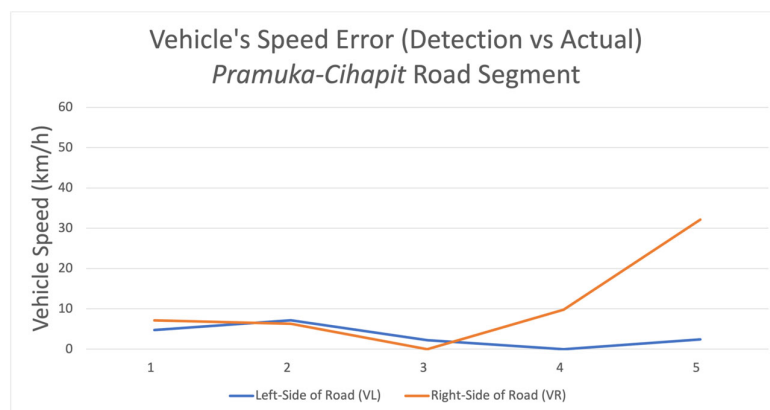
**Figure 14.** Average error value for vehicle counting on Pramuka–Cihapit road segment.

The average errors on the Pramuka–Cihapit road segment included two cars, two motorcycles, and one truck for both lanes. Meanwhile, there were no buses when the measurement was conducted. The error is caused by several reasons, such as: (1) the CCTV's resolution is too low and incorrectly detected the vehicle; (2) the vehicles are moving too fast; (3) there was a misclassification when detecting the type of vehicles since there are vehicles that have similar characteristics with other vehicles (e.g., small buses are considered as big buses).

Table 9 and Figure 15 show the result of vehicle speed calculation. The actual value is obtained using manual calculation based on the vehicle movement between two adjacent frames. Its velocity (km/h) is measured using the movement value (in pixels) in a specific time span (s). The average vehicle speed on the road is generalized from several types of vehicles by using the equivalency method, as explained in Equation (2). On the left and right sides of the road, it reached 3.308 and 11.088 km/h—the differences in detection and actual vehicle speed, respectively.

**Table 9.** Absolute error (km/h) for vehicle speed calculation on Pramuka–Cihapit road segment.

| No      | Left-Side of The Road (VL) |              |                | Right-Side of The Road (VR) |              |                |
|---------|----------------------------|--------------|----------------|-----------------------------|--------------|----------------|
|         | Detection Speed            | Actual Speed | Absolute Error | Detection Speed             | Actual Speed | Absolute Error |
| 1       | 21.16                      | 25.89        | 4.73           | 9.9                         | 17.06        | 7.16           |
| 2       | 21.34                      | 14.18        | 7.16           | 26.76                       | 20.43        | 6.33           |
| 3       | 4.65                       | 6.89         | 2.24           | 0                           | 0            | 0              |
| 4       | 0                          | 0            | 0              | 10.4                        | 20.22        | 9.82           |
| 5       | 7.33                       | 4.92         | 2.41           | 11.41                       | 43.54        | 32.13          |
| Average |                            |              | 3.308          |                             |              | 11.088         |



**Figure 15.** Error rate of vehicle speed calculation on Pramuka–Cihapit road segment.

#### Trunojoyo Road Segment

The location of CCTV in the Trunojoyo road segment is placed around 350 m from an intersection, in front of a hospital and near a supermarket. During the observation, it is common that the drivers on this road segment drive their vehicles at high speed to avoid the traffic light. Not only are vehicles passing the road at high speed, but also the CCTV resolution in this location is no better than at the Pramuka–Cihapit road segment. This condition makes the measurement result worse than the previous location. Table 10 shows the result of the number of vehicles calculated on the Trunojoyo road segment.

**Table 10.** Comparison between detection and actual result on Trunojoyo road segment.

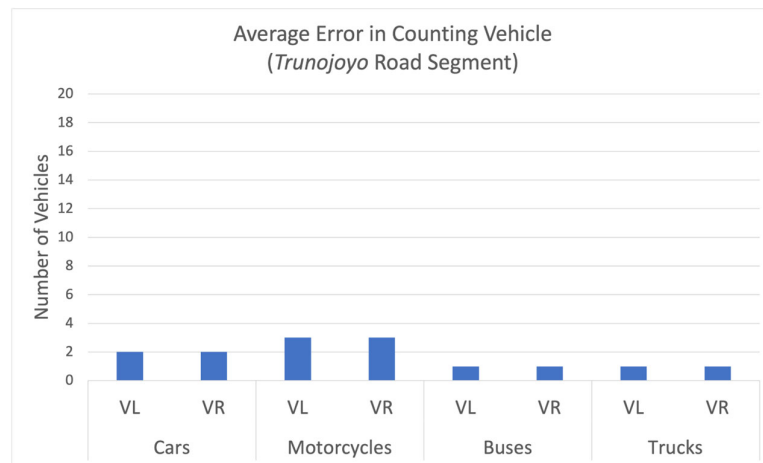
| Detection |    |            |    |     |    |       |    | Actual |    |            |    |     |    |       |    |
|-----------|----|------------|----|-----|----|-------|----|--------|----|------------|----|-----|----|-------|----|
| Car       |    | Motorcycle |    | Bus |    | Truck |    | Car    |    | Motorcycle |    | Bus |    | Truck |    |
| VL        | VR | VL         | VR | VL  | VR | VL    | VR | VL     | VR | VL         | VR | VL  | VR | VL    | VR |
| 2         | 3  | 1          | 1  | 0   | 0  | 0     | 0  | 1      | 4  | 2          | 1  | 0   | 0  | 0     | 0  |
| 4         | 2  | 0          | 1  | 0   | 0  | 1     | 0  | 4      | 1  | 0          | 5  | 0   | 0  | 0     | 0  |
| 0         | 1  | 0          | 0  | 0   | 0  | 0     | 0  | 0      | 4  | 1          | 1  | 0   | 0  | 0     | 0  |
| 7         | 1  | 2          | 0  | 0   | 0  | 0     | 0  | 5      | 4  | 2          | 2  | 0   | 0  | 0     | 0  |
| 3         | 0  | 1          | 1  | 0   | 0  | 1     | 0  | 6      | 0  | 2          | 2  | 0   | 0  | 0     | 0  |
| 4         | 2  | 1          | 1  | 0   | 0  | 0     | 0  | 6      | 4  | 1          | 1  | 0   | 0  | 0     | 0  |
| 0         | 1  | 4          | 1  | 0   | 0  | 0     | 1  | 1      | 0  | 4          | 1  | 0   | 0  | 1     | 0  |
| 2         | 0  | 2          | 1  | 0   | 0  | 0     | 0  | 4      | 2  | 4          | 6  | 0   | 0  | 0     | 0  |
| 2         | 0  | 5          | 0  | 0   | 0  | 0     | 0  | 1      | 3  | 16         | 5  | 0   | 0  | 0     | 0  |
| 1         | 1  | 2          | 1  | 0   | 0  | 0     | 0  | 4      | 1  | 3          | 6  | 0   | 0  | 0     | 0  |
| 1         | 3  | 0          | 0  | 0   | 0  | 0     | 0  | 3      | 5  | 2          | 3  | 0   | 0  | 0     | 0  |
| 1         | 1  | 3          | 0  | 0   | 0  | 0     | 0  | 1      | 2  | 6          | 3  | 0   | 0  | 0     | 0  |
| 4         | 1  | 1          | 2  | 0   | 0  | 0     | 0  | 6      | 3  | 4          | 2  | 0   | 0  | 1     | 0  |
| 2         | 2  | 0          | 1  | 0   | 0  | 0     | 0  | 3      | 5  | 0          | 7  | 0   | 0  | 0     | 0  |
| 1         | 3  | 3          | 0  | 0   | 0  | 0     | 0  | 2      | 4  | 7          | 4  | 0   | 0  | 0     | 0  |
| 6         | 3  | 0          | 1  | 0   | 0  | 0     | 0  | 7      | 3  | 3          | 2  | 0   | 0  | 0     | 0  |
| 2         | 2  | 0          | 0  | 0   | 1  | 0     | 0  | 4      | 5  | 5          | 0  | 0   | 0  | 0     | 0  |
| 2         | 3  | 1          | 1  | 0   | 0  | 0     | 0  | 3      | 3  | 2          | 4  | 0   | 0  | 0     | 0  |
| 1         | 3  | 3          | 0  | 0   | 0  | 0     | 0  | 1      | 4  | 14         | 0  | 0   | 0  | 0     | 0  |
| 1         | 5  | 1          | 0  | 0   | 0  | 0     | 0  | 2      | 6  | 3          | 3  | 0   | 0  | 0     | 0  |
| 7         | 1  | 1          | 0  | 1   | 0  | 0     | 0  | 9      | 3  | 2          | 4  | 0   | 0  | 0     | 0  |
| 5         | 1  | 0          | 1  | 0   | 0  | 1     | 0  | 8      | 4  | 2          | 1  | 0   | 0  | 0     | 0  |
| 0         | 0  | 0          | 0  | 0   | 0  | 1     | 2  | 1      | 3  | 0          | 1  | 0   | 0  | 1     | 0  |
| 1         | 1  | 0          | 1  | 0   | 0  | 1     | 0  | 4      | 1  | 1          | 6  | 0   | 0  | 1     | 0  |

|   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | 2 | 3 | 0 | 0 | 0 | 0 | 0 | 1 | 3 | 6 | 1 | 0 | 0 | 0 | 0 |
| 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 2 | 1 | 3 | 0 | 0 | 0 | 0 |
| 4 | 2 | 1 | 2 | 0 | 0 | 0 | 0 | 5 | 3 | 1 | 6 | 0 | 0 | 0 | 0 |
| 5 | 1 | 0 | 2 | 0 | 0 | 1 | 0 | 9 | 0 | 2 | 7 | 0 | 0 | 0 | 0 |
| 6 | 2 | 0 | 1 | 0 | 0 | 0 | 0 | 7 | 2 | 1 | 1 | 0 | 0 | 0 | 0 |
| 2 | 1 | 3 | 0 | 0 | 0 | 0 | 0 | 3 | 1 | 4 | 1 | 0 | 0 | 0 | 0 |

The value of average error in this area is slightly higher than in the previous road segment. Figure 16 shows the comparison value for each lane based on the vehicle types. As shown in Table 11, on both lanes, the number of cars and trucks have similar values to the measurement on Pramuka–Cihapit road segment. However, the average number of differences between motorcycles and buses is three and one for both lanes. The characteristics of this road are almost similar to the previous road segment, which has a low intensity of heavy vehicles on the road.

**Table 11.** Error rate in vehicle counting on Trunojoyo road segment.

| Testing | Car |    | Motorcycle |    | Bus |    | Truck |    |
|---------|-----|----|------------|----|-----|----|-------|----|
|         | VL  | VR | VL         | VR | VL  | VR | VL    | VR |
| 1       | 1   | 1  | 1          | 0  | 0   | 0  | 0     | 0  |
| 2       | 0   | 1  | 0          | 4  | 0   | 0  | 1     | 0  |
| 3       | 0   | 3  | 1          | 1  | 0   | 0  | 0     | 0  |
| 4       | 2   | 3  | 0          | 2  | 0   | 0  | 0     | 0  |
| 5       | 3   | 0  | 1          | 1  | 0   | 0  | 1     | 0  |
| 6       | 2   | 2  | 0          | 0  | 0   | 0  | 0     | 0  |
| 7       | 1   | 1  | 0          | 0  | 0   | 0  | 1     | 1  |
| 8       | 2   | 2  | 2          | 5  | 0   | 0  | 0     | 0  |
| 9       | 1   | 3  | 11         | 5  | 0   | 0  | 0     | 0  |
| 10      | 3   | 0  | 1          | 5  | 0   | 0  | 0     | 0  |
| 11      | 2   | 2  | 2          | 3  | 0   | 0  | 0     | 0  |
| 12      | 0   | 1  | 3          | 3  | 0   | 0  | 0     | 0  |
| 13      | 2   | 2  | 3          | 0  | 0   | 0  | 1     | 0  |
| 14      | 1   | 3  | 0          | 6  | 0   | 0  | 0     | 0  |
| 15      | 1   | 1  | 4          | 4  | 0   | 0  | 0     | 0  |
| 16      | 1   | 0  | 3          | 1  | 0   | 0  | 0     | 0  |
| 17      | 2   | 3  | 5          | 0  | 0   | 1  | 1     | 0  |
| 18      | 1   | 0  | 1          | 3  | 0   | 0  | 0     | 0  |
| 19      | 0   | 1  | 11         | 0  | 0   | 0  | 0     | 0  |
| 20      | 1   | 1  | 2          | 3  | 0   | 0  | 0     | 0  |
| 21      | 2   | 2  | 1          | 4  | 1   | 0  | 0     | 0  |
| 22      | 3   | 3  | 2          | 0  | 0   | 0  | 1     | 0  |
| 23      | 1   | 3  | 0          | 1  | 0   | 0  | 0     | 2  |
| 24      | 3   | 0  | 1          | 5  | 0   | 0  | 0     | 0  |
| 25      | 1   | 1  | 3          | 1  | 0   | 0  | 0     | 0  |
| 26      | 1   | 1  | 0          | 3  | 0   | 0  | 0     | 0  |
| 27      | 1   | 1  | 0          | 4  | 0   | 0  | 0     | 0  |
| 28      | 4   | 1  | 2          | 5  | 0   | 0  | 1     | 0  |
| 29      | 1   | 0  | 1          | 0  | 0   | 0  | 0     | 0  |
| 30      | 1   | 0  | 1          | 1  | 0   | 0  | 0     | 0  |
| Average | 2   | 2  | 3          | 3  | 1   | 1  | 1     | 1  |

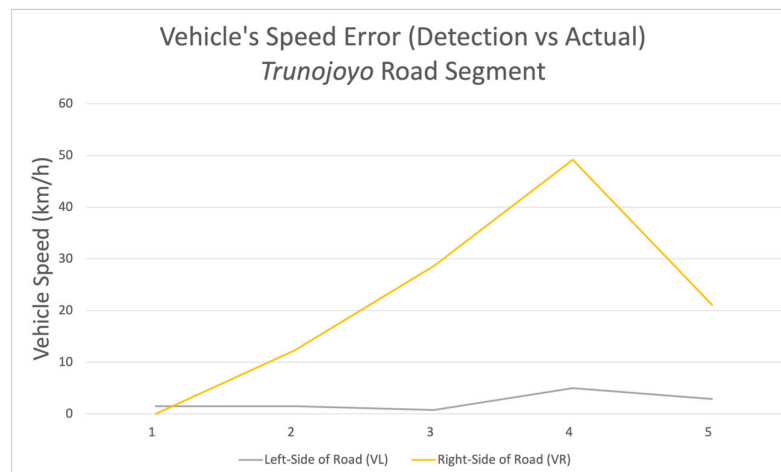


**Figure 16.** Average error value for vehicle counting on Trunojoyo road segment.

Based on the observation in the CCTV, the object detected on the left side is relatively clear since it is moving slowly. Meanwhile, on the right side, a lot of vehicles are moving too fast, meaning that the vehicle speed measurement has a greater error rate than in the other lane. As shown in Table 12 and Figure 17, the differences in vehicle speed on the left and right sides are 2.316 and 22.222 km/h, respectively.

**Table 12.** Error rate (km/h) for vehicle speed calculation on Trunojoyo road segment.

| No      | Left-Side of The Road (VL) |              |                | Right-Side of the Road (VR) |              |                |
|---------|----------------------------|--------------|----------------|-----------------------------|--------------|----------------|
|         | Detection Speed            | Actual Speed | Absolute Error | Detection Speed             | Actual Speed | Absolute Error |
| 1       | 21.29                      | 22.78        | 1.49           | 0                           | 0            | 0              |
| 2       | 17.64                      | 19.12        | 1.48           | 50                          | 37.7         | 12.3           |
| 3       | 21.23                      | 20.49        | 0.74           | 9.64                        | 38.23        | 28.59          |
| 4       | 12.19                      | 17.17        | 4.98           | 4.28                        | 53.42        | 49.14          |
| 5       | 17.93                      | 20.82        | 2.89           | 10.21                       | 31.29        | 21.08          |
| Average |                            |              | 2.316          |                             |              | 22.222         |



**Figure 17.** Error rate of vehicle speed calculation on Trunojoyo road segment.

### Merdeka-Aceh Intersection

The CCTV on the Merdeka-Aceh intersection is easily moved by the operator since it is a PTZ (Pan-Tilt-Zoom) CCTV system. However, overall, it mostly observed the condition of the Merdeka Road Segment. This road follows a one-way direction to its intersection. Table 13 shows the vehicle measurement results on this road.

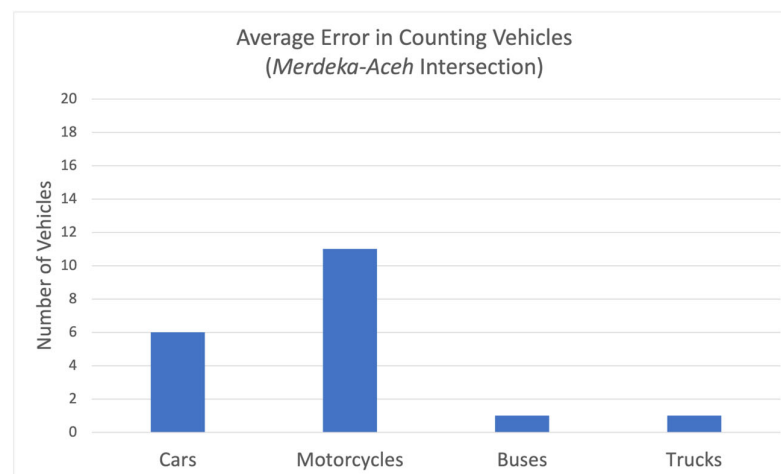
**Table 13.** Comparison between detection and actual result on Merdeka-Aceh Intersection.

| Detection |            |     |       | Actual |            |     |       |
|-----------|------------|-----|-------|--------|------------|-----|-------|
| Car       | Motorcycle | Bus | Truck | Car    | Motorcycle | Bus | Truck |
| 26        | 0          | 1   | 2     | 23     | 18         | 0   | 0     |
| 8         | 2          | 1   | 2     | 7      | 34         | 0   | 0     |
| 5         | 0          | 0   | 0     | 6      | 16         | 0   | 0     |
| 5         | 0          | 0   | 0     | 8      | 8          | 0   | 0     |
| 25        | 3          | 1   | 1     | 21     | 27         | 0   | 0     |
| 5         | 3          | 0   | 1     | 13     | 15         | 0   | 0     |
| 16        | 2          | 0   | 0     | 21     | 12         | 0   | 0     |
| 3         | 5          | 0   | 0     | 26     | 15         | 0   | 0     |
| 4         | 4          | 0   | 0     | 10     | 16         | 0   | 0     |
| 13        | 3          | 0   | 0     | 23     | 17         | 0   | 0     |
| 10        | 6          | 1   | 1     | 16     | 8          | 1   | 0     |
| 6         | 1          | 0   | 2     | 17     | 4          | 0   | 0     |
| 5         | 2          | 0   | 0     | 5      | 10         | 0   | 0     |
| 5         | 4          | 2   | 0     | 15     | 10         | 0   | 0     |
| 5         | 9          | 0   | 0     | 6      | 13         | 0   | 0     |
| 26        | 5          | 0   | 3     | 23     | 17         | 0   | 0     |
| 5         | 7          | 0   | 2     | 12     | 18         | 0   | 0     |
| 10        | 7          | 1   | 0     | 19     | 22         | 0   | 0     |
| 9         | 3          | 2   | 0     | 15     | 17         | 0   | 0     |
| 3         | 5          | 0   | 0     | 7      | 6          | 0   | 0     |
| 0         | 5          | 0   | 0     | 7      | 10         | 0   | 0     |
| 4         | 4          | 0   | 0     | 5      | 11         | 0   | 0     |
| 6         | 13         | 0   | 0     | 8      | 19         | 0   | 0     |
| 6         | 4          | 0   | 2     | 7      | 12         | 0   | 0     |
| 23        | 5          | 1   | 0     | 22     | 18         | 0   | 1     |
| 5         | 2          | 0   | 0     | 10     | 7          | 0   | 0     |
| 11        | 5          | 0   | 0     | 14     | 9          | 0   | 0     |
| 15        | 4          | 1   | 2     | 20     | 17         | 0   | 1     |
| 7         | 7          | 1   | 2     | 14     | 18         | 0   | 0     |
| 13        | 1          | 4   | 2     | 19     | 20         | 0   | 0     |

There are several obstacles from the CCTV's point of view, such as streetlights that appeared to be in the middle of the road and special places for motorcycles to wait until the traffic lights turn green. These conditions make the vehicles hard to detect. Moreover, the range of vehicles when stopped during the red light is too tight, which also makes the vehicles harder to detect. Although the average error value in counting the number of cars on this road is smaller than the other roads, it has the highest average error value in measuring the number of motorcycles on other roads. As shown in Table 14, the error value for counting the number of cars and motorcycles reached 6 and 11, respectively. Figure 18 shows a comparison between the average error value based on the type of vehicle.

**Table 14.** Error rate in vehicle counting on Merdeka-Aceh Intersection.

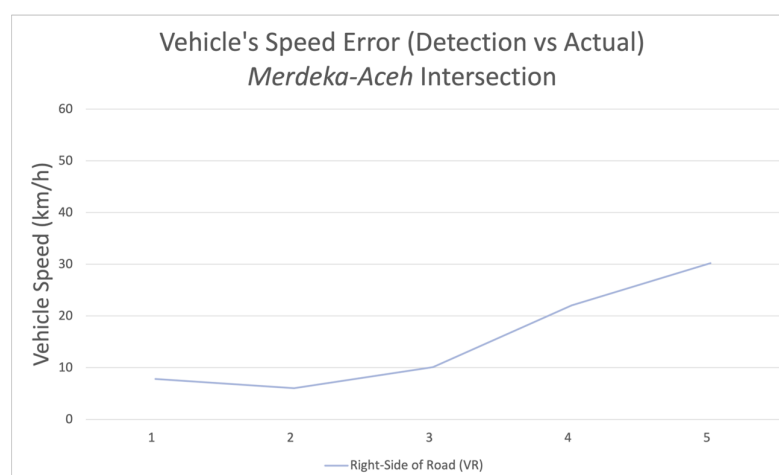
| Testing | Car | Motorcycle | Bus | Truck |
|---------|-----|------------|-----|-------|
| 1       | 3   | 18         | 1   | 2     |
| 2       | 1   | 32         | 1   | 2     |
| 3       | 1   | 16         | 0   | 0     |
| 4       | 3   | 8          | 0   | 0     |
| 5       | 4   | 24         | 1   | 1     |
| 6       | 8   | 12         | 0   | 1     |
| 7       | 5   | 10         | 0   | 0     |
| 8       | 23  | 10         | 0   | 0     |
| 9       | 6   | 12         | 0   | 0     |
| 10      | 10  | 14         | 0   | 0     |
| 11      | 6   | 2          | 0   | 1     |
| 12      | 11  | 3          | 0   | 2     |
| 13      | 0   | 8          | 0   | 0     |
| 14      | 10  | 6          | 2   | 0     |
| 15      | 1   | 4          | 0   | 0     |
| 16      | 3   | 12         | 0   | 3     |
| 17      | 7   | 11         | 0   | 2     |
| 18      | 9   | 15         | 1   | 0     |
| 19      | 6   | 14         | 2   | 0     |
| 20      | 4   | 1          | 0   | 0     |
| 21      | 7   | 5          | 0   | 0     |
| 22      | 1   | 7          | 0   | 0     |
| 23      | 2   | 6          | 0   | 0     |
| 24      | 1   | 8          | 0   | 2     |
| 25      | 1   | 13         | 1   | 1     |
| 26      | 5   | 5          | 0   | 0     |
| 27      | 3   | 4          | 0   | 0     |
| 28      | 5   | 13         | 1   | 1     |
| 29      | 7   | 11         | 1   | 2     |
| 30      | 6   | 19         | 4   | 2     |
| Average | 6   | 11         | 1   | 1     |

**Figure 18.** Average error value for vehicle counting on Merdeka-Aceh.

Since there is an attractive place near the intersection, the number of vehicles on this road is high. This road is also a primary road in Bandung, Indonesia. On average, the error rate in measuring the vehicle speed at the Merdeka-Aceh intersection is 15.26 km/h, as shown in Table 15 and Figure 19.

**Table 15.** Absolute error (km/h) for vehicle speed calculation on Merdeka-Aceh Intersection.

| No      | Right-Side of the Road (VR) |              |                |
|---------|-----------------------------|--------------|----------------|
|         | Detected Speed              | Actual Speed | Absolute Error |
| 1       | 13.67                       | 21.51        | 7.84           |
| 2       | 6.64                        | 12.71        | 6.07           |
| 3       | 20.26                       | 30.37        | 10.11          |
| 4       | 14.1                        | 36.17        | 22.07          |
| 5       | 16.64                       | 46.85        | 30.21          |
| Average |                             |              | 15.26          |



**Figure 19.** Average error rate of vehicle speed calculation on Merdeka-Aceh Intersection.

#### 4.2. The Measurement of Traffic Condition Category

In the implementation, the methods in this paper were used to determine the category of traffic conditions according to the saturation degree. This requires information such as days, times, and weather conditions to define its condition. For the area covered by CCTV, the width of the road is also needed to measure its base saturation. As explained in Equation (11), the input when categorizing the traffic condition is the saturation degree. Meanwhile, TomTom digital maps work as a subsidiary source to gather road situations based on vehicle speed.

##### 4.2.1. Traffic Conditions Based on CCTV

The main difference in calculating the traffic condition in an intersection and a road segment is the traffic light cycle and its green time. In general, the green time ( $g$ ) is in a range of 40–50 s and the whole cycle ( $c$ ) takes 80–120 s. These values are only applied when the CCTV is surveilling an intersection. When it is placed on a road segment in which there is no traffic light, the comparison between the green time and its cycles is set to 1.

Table 16 shows the result of data collection using CCTV. The number of every vehicle type is collected, and, later, it is processed so that the number of vehicles becomes equivalent. The traffic condition is stored in each road segment dataset along with the other information collected.

**Table 16.** Traffic condition measurement results based on CCTV.

| Location          | Cars | Motorcycles | Buses | Trucks | Road Width | $g$ | $c$ | $Q_{hour}$ | $S$      | $DS$  | Traffic Condition |
|-------------------|------|-------------|-------|--------|------------|-----|-----|------------|----------|-------|-------------------|
| Juanda–Merdeka    | 12   | 10          | 0     | 1      | 14.8       | 70  | 100 | 5508       | 10,389.6 | 0.757 | 3                 |
| Trunojoyo–Merdeka | 13   | 2           | 0     | 0      | 5.6        | 50  | 120 | 4824       | 3931.2   | 2.945 | 3                 |
| Merdeka–Trunojoyo | 2    | 1           | 0     | 0      | 5.6        | 50  | 120 | 792        | 3931.2   | 0.484 | 1                 |
| Pramuka–Cihapit   | 2    | 2           | 0     | 0      | 4.8        | 1   | 1   | 864        | 3369.6   | 0.256 | 1                 |
| Cihapit–Pramuka   | 2    | 0           | 0     | 0      | 4.8        | 1   | 1   | 720        | 3369.6   | 0.214 | 0                 |
| Trunojoyo–Merdeka | 9    | 4           | 1     | 4      | 5.6        | 50  | 120 | 5868       | 3931.2   | 3.582 | 3                 |

The time span used for observations using CCTV is 10 s. A conversion is required to adjust the hourly traffic flow.  $Q_{hour}$  is adjusted based on its observation time ( $Q_{hour} = Q_{10\ sec} \times 360$ ). Meanwhile, to define the value of the saturated flow, it is measured according to its base saturation from each road and its factors. This value completes the saturation degree calculation.

This process is performed periodically for several weeks to create the traffic condition dataset for each road segment. The dataset that has already been collected can be used to predict future traffic conditions.

#### 4.2.2. Traffic Condition Based on TomTom Digital Maps

Table 17 shows the calculation result using the TomTom digital maps. As shown in the table, the values of the current and free flow speed are needed to define the degree of saturation and the traffic condition. Whenever the current speed and the free flow speed show similar values, the degree of saturation is 0. Otherwise, the line equation is applied to determine its degree.

**Table 17.** The calculation result of saturation degree and traffic condition using TomTom.

| Location           | Traffic Information                   | $DS$ | Traffic Condition |
|--------------------|---------------------------------------|------|-------------------|
| Lombok–Pramuka     | currentSpeed: 27<br>freeFlowSpeed: 35 | 0.69 | 2                 |
| Seram–Saparua      | currentSpeed: 27<br>freeFlowSpeed: 27 | 0    | 0                 |
| Gudang Utara–Laswi | currentSpeed: 26<br>freeFlowSpeed: 35 | 0.77 | 3                 |

Based on the MKJI, the maximum saturation degree that can be calculated using the line equation is limited to 80%. Thus, if there is a saturation degree over 80%, it is equal to very heavy traffic. This method was applied to all road segments that were observed using TomTom digital maps.

#### 4.3. The Collection of Weather Information

As shown in Figure 20, the whole traffic information is collected for the Pramuka–Lombok road segment. Based on the server's response, the weather data indicated an “overcast cloud” at the observation time. Meanwhile, the other subsidiary data that were collected show its temperature and humidity.

The main information used in this process is the weather data itself. The variances of weather provided by the OpenWeather are categorized as “Thunderstorm”, “Drizzle”, “Rain”, “Snow”, “Atmosphere”, “Clear”, and “Clouds”. Since Indonesia is a tropical country, we chose several unused weather conditions that only occurred in Indonesia, such as “Rain”, “Clear”, and “Clouds”. We also reduced its subcategories into nine weather conditions, namely “clear sky”, “few clouds”, “scattered clouds”, “broken clouds”, “overcast clouds”, “light rain”, “moderate rain”, “heavy intensity rain”, and

“very heavy rain”. These conditions were assigned a numerical value from 1 to 9, described from the best to the worst conditions.

```
=====
===== Data : {Source : , Pramuka , Destination : Lombok }=====
Traffic Condition : 2
Weather Condition : overcast clouds
Temperature : 23.56
Humidity : 88
Road Length : 819
Heterogenity : 1
Current Speed : 27
Free Flow Speed : 35
=====
```

**Figure 20.** Road information collection.

#### 4.4. Normalization of Traffic Condition Dataset

By the time the road information is collected, the conversion of the data structure must be complete to make it uniform. The information of “Days” consists of one of “Monday” to “Sunday”, which comes in the form of a String. This must be converted into numerical values; therefore, this information was assigned a range of values from 0 to 6 for each day (Monday to Sunday).

The information of “Times” is spread into 24 h; however, in our dataset, it is limited to the rush hour and non-rush hour time. Therefore, the information of time in the dataset is converted as rush hour (1) and a non-rush hour (0). This is categorized based on the Indonesian Department of Transportation, for which the rush hours in Bandung are 7–9 a.m. and 4–7 p.m. Whenever a time is collected in the range of rush hour time, it is set as (1) and the others are set as (0).

Weather data come in the form of a String format. We labeled it sequentially based on the weather level; the worst weather has the greatest numerical value. All the weather levels are clear sky (1), few clouds (2), scattered clouds (3), broken clouds (4), overcast clouds (5), light rain (6), moderate rain (7), heavy intensity rain (8), and very heavy rain (9). The other weather information and the traffic conditions were not converted since they already come in numerical form.

Whenever the data were in numerical form, the normalization process was initiated. This was performed to unify the range of the values of each piece of information. It is performed by implementing the MinMax normalization method for each collected piece of information, except for the traffic condition and days. The value of traffic conditions was used as a class in predicting the traffic condition. In the end, each piece of information had the same range (from 0 to 1). Tables 18 and 19 show the results of the dataset collected on the “AhmadYani-Supratman” and “Menado2-GudangUtara4” road segments, respectively.

**Table 18.** The dataset collection result for AhmadYani-Supratman road segment.

| Days | Rush Hour | Weather | Temperature | Humidity | Traffic Condition |
|------|-----------|---------|-------------|----------|-------------------|
| 4    | 0         | 0.5     | 0.74        | 0.19     | 3                 |
| 1    | 0         | 0.75    | 0.72        | 0.43     | 3                 |
| 6    | 0         | 0.62    | 0.71        | 0.43     | 3                 |
| 3    | 1         | 0.5     | 0.78        | 0.09     | 2                 |
| 4    | 0         | 0.5     | 0.74        | 0.19     | 3                 |
| 0    | 1         | 0.62    | 0.76        | 0.55     | 0                 |
| 4    | 0         | 0.25    | 0.78        | 0.11     | 3                 |

**Table 19.** The dataset collection result for Menado2-GudangUtara4 road segment.

| Days | Rush Hour | Weather | Temperature | Humidity | Traffic Condition |
|------|-----------|---------|-------------|----------|-------------------|
| 2    | 1         | 0.5     | 0.76        | 0.55     | 0                 |
| 2    | 1         | 0.25    | 0.76        | 0.55     | 3                 |
| 1    | 0         | 0.62    | 0.72        | 0.43     | 2                 |
| 1    | 0         | 0.75    | 0.72        | 0.43     | 3                 |
| 6    | 0         | 0.5     | 0.76        | 0.55     | 2                 |
| 0    | 0         | 0.5     | 0.73        | 0.35     | 3                 |
| 4    | 1         | 0.25    | 0.78        | 0.11     | 0                 |

## 5. Conclusions

In this paper, the traffic condition is defined by using one of the proposed methods, namely object detection based on CCTVs stream and TomTom digital maps which only applied to the road that has no CCTV coverage. This paper provides the method of dataset collection for heterogeneous traffic flow located in Indonesia. This is the first dataset for Indonesian traffic based on a real situation.

The average error rate in determining the number of vehicles is in the range of 0–2 and 2–3 vehicles for cars and motorcycles, respectively, and 0–1 vehicles for others. This error value is caused by (1) the low resolution of the public CCTV systems, which makes the objects blurry, (2) some obstacles interfering with the CCTV system's view, and (3) the misdetection of the vehicle type. The average error of the vehicle speed measurement on the left-side of the road is in the range of 2–3 km/h. Meanwhile, for the other side, its average error is between 11.088 and 22.222 km/h for all observed road segments. On several roads, the vehicles are moving too fast, making the frames from the CCTV streams blurry and making our proposed framework calculate their velocity incorrectly.

We collected the traffic information for 265 road segments in the observed area. These datasets will be used as training data for predicting traffic conditions. Later, based on the predicted traffic condition, vehicle speed, and other road criteria, the weight of the road segment is calculated so that the driver can decide the best route with minimum weight to their destination.

For future study, the implementation of the object detection method should be performed with various detection algorithms so that the algorithm with the lowest value of error can be determined. Since public CCTV systems have a bad resolution, it is hard to implement computer vision (object detection, etc.) using this type of CCTV. The implementation of higher-resolution cameras (or private CCTV) would be much more effective.

**Author Contributions:** Conceptualization, S.M.N. and E.H.; methodology, S.M.N., E.H., K.K. and R.Y.; software, S.M.N.; validation, S.M.N. and R.Y.; formal analysis, S.M.N., E.H. and K.K.; investigation, S.M.N.; resources, S.M.N.; data curation, S.M.N. and R.Y.; writing—original draft preparation, S.M.N.; writing—review and editing, S.M.N.; visualization, S.M.N.; supervision, E.H. and K.K.; project administration, S.M.N.; funding acquisition, S.M.N. All authors have read and agreed to the published version of the manuscript.

**Funding:** This research received no external funding.

**Institutional Review Board Statement:** Not applicable.

**Informed Consent Statement:** Not applicable.

**Data Availability Statement:** The heterogeneous traffic condition dataset collected during the current study is available online, <http://dx.doi.org/10.6084/m9.figshare.21154486> (access on 1 February 2023).

**Conflicts of Interest:** The authors declare no conflict of interest. The funders had no role in the design of the study; in the collection, analyses, or interpretation of data; in the writing of the manuscript; or in the decision to publish the results.

## References

1. BPS. *Perkembangan Jumlah Kendaraan Bermotor Menurut Jenis, 1949–2016*; BPS: Jakarta, Indonesia, 2018.
2. Moyo, T.; Mbatha, S.; Aderibigbe, O.-O.; Gumbo, T.; Musonda, I. Assessing Spatial Variations of Traffic Congestion Using Traffic Index Data in a Developing City: Lessons from Johannesburg, South Africa. *Sustainability* **2022**, *14*, 8809. <https://doi.org/10.3390/su14148809>.
3. Redmon, J.; Divvala, S.; Girshick, R.; Farhadi, A. You Only Look Once: Unified, Real-Time Object Detection. In Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 27–30 June 2016.
4. Lin, T.Y.; Goyal, P.; Girshick, R.; He, K.; Dollar, P. Focal Loss for Dense Object Detection. *IEEE Trans. Pattern Anal. Mach. Intell.* **2020**, *42*, 318–327.
5. Jiang, X.; Gao, T.; Zhu, Z.; Zhao, Y. Real-Time Face Mask Detection Method Based on Yolov3. *Electronics* **2021**, *10*, 837. <https://doi.org/10.3390/electronics10070837>.
6. Chun, L.Z.; Dian, L.; Zhi, J.Y.; Jing, W.; Zhang, C. YOLOv3: Face Detection in Complex Environments. *Int. J. Comput. Intell. Syst.* **2020**, *13*, 1153–1160. <https://doi.org/10.2991/ijcis.d.200805.002>.
7. Wang, H.; Zhang, Z. Text Detection Algorithm Based on Improved YOLOv3. In Proceedings of the 2019 IEEE 9th International Conference on Electronics Information and Emergency Communication (ICEIEC), Beijing, China, 12–14 July 2019; pp. 147–150.
8. Xiao, L.; Zhou, P.; Xu, K.; Zhao, X. Multi-Directional Scene Text Detection Based on Improved Yolov3. *Sensors* **2021**, *21*, 4870. <https://doi.org/10.3390/s21144870>.
9. Marques, R.; Ribeiro, T.; Lopes, G.; Ribeiro, A. YOLOv3: Traffic Signs & Lights Detection and Recognition for Autonomous Driving. In Proceedings of the 14th International Conference on Agents and Artificial Intelligence, Vienna, Austria, 3–5 February 2022; Volume 3, pp. 818–826. <https://doi.org/10.5220/0010914100003116>.
10. Xiang, N.; Cao, Z.; Wang, Y.; Jia, Q. A Real-Time Vehicle Traffic Light Detection Algorithm Based on Modified YOLOv3. In Proceedings of the 2021 IEEE 4th International Conference on Electronics Technology (ICET), Chengdu, China, 7–10 May 2021; pp. 844–850.
11. Salam, H.; Jaleel, H.; Hameedi, S. You Only Look Once (YOLOv3): Object Detection and Recognition for Indoor Environment. *Multicult. Educ.* **2021**, *7*, 174. <https://doi.org/10.5281/zenodo.4906284>.
12. Prabhu, S.; Khopkar, V.; Nivendkar, S.; Satpute, O.; Jyotinagar, V. Object Detection and Classification Using GPU Acceleration. *Adv. Intell. Syst. Comput.* **2020**, *1108*, 161–170. [https://doi.org/10.1007/978-3-030-37218-7\\_18](https://doi.org/10.1007/978-3-030-37218-7_18).
13. Warsi, A.; Abdullah, M.; Husen, M.N.; Yahya, M.; Khan, S.; Jawaid, N. Gun Detection System Using Yolov3. In Proceedings of the 2019 IEEE 6th International Conference on Smart Instrumentation, Measurement and Application (ICSIMA), Kuala Lumpur, Malaysia, 27–29 August 2019; pp. 1–4. <https://doi.org/10.1109/ICSIMA47653.2019.9057329>.
14. Nasution, S.M.; Husni, E.; Kuspriyanto, K.; Yusuf, R.; Yahya, B.N. Contextual Route Recommendation System in Heterogeneous Traffic Flow. *Sustainability* **2021**, *13*, 13191. <https://doi.org/10.3390/su132313191>.
15. Susilo, B.H.; Imanuel, I. Traffic Congestion Analysis Using Travel Time Ratio and Degree of Saturation on Road Sections in Palembang, Bandung, Yogyakarta, and Surakarta. *MATEC Web Conf.* **2018**, *181*, 06010.
16. PT. Bina Karya (Persero). *Manual Kapasitas Jalan Indonesia*; PT. Bina Karya (Persero): Jakarta, Indonesia, 1997.
17. Juniarta, I.; Negara, I.; Wikrama, A. Penentuan Nilai Ekuivalensi Mobil Penumpang Pada Ruas Jalan Perkotaan. *J. Ilm. Elektron. Infrastruktur Tek. Sipil* **2012**, *1*, 1–7.
18. Yulipriyono, E.E.; Purwanto, D. Perubahan Nilai Ekuivalensi Mobil Penumpang Akibat Perubahan Karakteristik Operasional Kendaraan Di Jalan Kota Semarang. *Media Komun. Tek. Sipil* **2017**, *23*, 69. <https://doi.org/10.14710/mkts.v23i1.12517>.
19. Sugeng, R. *Rekayasa Dan Manajemen Lalu Lintas: Teori Dan Aplikasi*, 1st ed.; LeutikaPrio: Yogyakarta, Indonesia, 2014.
20. Munawar, A. *Manajemen Lalu Lintas Perkotaan*; Beta Offset: Yogyakarta, Indonesia, 2004.
21. Rahayu, G.; Rosyidi, S.A.P.; Munawar, A. Analisis Arus Jenuh Dan Panjang Antrian Pada Simpang Bersinyal: Studi Kasus Di Jalan Dr. Sutomo-Suryopranoto, Yogyakarta. *J. Ilm. Semesta Tek.* **2009**, *12*, 99–108.
22. Bester, C.J.; Meyers, W.L. Saturation Flow Rates. In Proceedings of the SATC 2007—26th Annual Southern African Transport “The Challenges of Implementing Policy?”, Pretoria, South Africa, 9–12 July 2007; pp. 560–568. <https://doi.org/10.1061/9780784404645.ch07>.
23. Aoyama, E.; Yoshioka, K.; Shimokawa, S.; Morita, H. Estimating Saturation Flow Rates at Signalized Intersections in Japan. *Asian Transp. Stud.* **2020**, *6*, 100015. <https://doi.org/10.1016/j.eastsj.2020.100015>.
24. Nasution, S.M.; Husni, E.; Kuspriyanto. The Effect of Heterogeneous Traffic Flow on The Transportation System. In Proceedings of the International Conference on Electrical Engineering and Computer Science, Bali, Indonesia, 13 November 2018; Volume 1.
25. Irawan, M.Z.; Sumi, T.; Munawar, A. Implementation of the 1997 Indonesian Highway Capacity Manual (MKJI) Volume Delay Function. *Proc. East. Asia Soc. Transp. Stud.* **2009**, *7*, 1–11.
26. Treiber, M.; Kesting, A. Automatic and Efficient Driving Strategies While Approaching a Traffic Light. In Proceedings of the 17th International IEEE Conference on Intelligent Transportation Systems (ITSC), Qingdao, China, 8–11 October 2014.
27. Lin, T.Y.; Maire, M.; Belongie, S.; Hays, J.; Perona, P.; Ramanan, D.; Dollár, P.; Zitnick, C.L. Microsoft COCO: Common Objects in Context. *Lect. Notes Comput. Sci.* **2014**, *8693*, 740–755. [https://doi.org/10.1007/978-3-319-10602-1\\_48](https://doi.org/10.1007/978-3-319-10602-1_48).
28. Szarvas, M.; Yoshizawa, A.; Yamamoto, M.; Ogata, J. Pedestrian Detection with Convolutional Neural Networks. In Proceedings of the IEEE Intelligent Vehicles Symposium, Las Vegas, NV, USA, 6–8 June 2005; pp. 224–229. <https://doi.org/10.1109/IVS.2005.1505106>.

29. De Smedt, F.; Goedemé, T. Open Framework for Combined Pedestrian Detection. In Proceedings of the VISAPP 2015—10th International Conference on Computer Vision Theory and Applications, Berlin, Germany, 11–14 March 2015; Volume 2, pp. 551–558. <https://doi.org/10.5220/0005359205510558>.
30. Hasan, I.; Liao, S.; Li, J.; Akram, S.U.; Shao, L. Generalizable Pedestrian Detection: The Elephant in the Room. In Proceedings of the 2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Nashville, TN, USA, 20–25 June 2021; pp. 11323–11332. <https://doi.org/10.1109/CVPR46437.2021.01117>.
31. Owayjan, M.; Dergham, A.; Haber, G.; Fakih, N.; Hamoush, A.; Abdo, E. Face Recognition Security System. *Lect. Notes Electr. Eng.* **2015**, *312*, 343–348. [https://doi.org/10.1007/978-3-319-06764-3\\_42](https://doi.org/10.1007/978-3-319-06764-3_42).
32. Dirgantara, F.M.; Wicaksa, D.P. Design of Face Recognition Security System on Public Spaces. *J. Electr. Electron. Inf. Commun. Technol.* **2022**, *4*, 6. <https://doi.org/10.20961/jeeict.4.1.60409>.
33. Li, X.; Wang, W.; Jiang, S.; Huang, Q.; Gao, W. Fast and Effective Text Detection. In Proceedings of the 2008 15th IEEE International Conference on Image Processing, San Diego, CA, USA, 12–15 October 2008; pp. 969–972. <https://doi.org/10.1109/ICIP.2008.4711918>.
34. Van Staden, J.; Brown, D. An Evaluation of YOLO-Based Algorithms for Hand Detection in the Kitchen. In Proceedings of the 2021 International Conference on Artificial Intelligence, Big Data, Computing and Data Communication Systems (icABCD), Durban, South Africa, 5–6 August 2021. <https://doi.org/10.1109/icABCD51485.2021.9519307>.
35. Chayeb, A.; Ouadah, N.; Tobal, Z.; Lakrouf, M.; Azouaoui, O. HOG Based Multi-Object Detection for Urban Navigation. In Proceedings of the 17th International IEEE Conference on Intelligent Transportation Systems (ITSC), Qingdao, China, 8–11 October 2014; pp. 2962–2967. <https://doi.org/10.1109/ITSC.2014.6958165>.
36. Zhang, L.; Wang, H.; Wang, X.; Chen, S.; Wang, H.; Zheng, K.; Wang, H. Vehicle Object Detection Based on Improved RetinaNet. *J. Phys. Conf. Ser.* **2021**, *1757*, 012070. <https://doi.org/10.1088/1742-6596/1757/1/012070>.
37. 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. <https://doi.org/10.1109/TPAMI.2016.2577031>.
38. Liu, W.; Anguelov, D.; Erhan, D.; Szegedy, C.; Reed, S.; Fu, C.-Y.; Berg, A.C. SSD: Single Shot MultiBox Detector. In *Computer Vision—ECCV 2016, Proceedings of the 14th European Conference, Amsterdam, The Netherlands, 11–14 October 2016*; Springer: Berlin/Heidelberg, Germany; pp. 398–413. <https://doi.org/10.1007/978-3-319-46448-0>.
39. Redmon, J.; Farhadi, A. YOLOv3: An Incremental Improvement. *arXiv* **2018**, arXiv:1804.02767.
40. Pang, L.; Liu, H.; Chen, Y.; Miao, J. Real-Time Concealed Object Detection from Passive Millimeter Wave Images Based on the YOLOV3 Algorithm. *Sensors* **2020**, *20*, 1678. <https://doi.org/10.3390/s20061678>.
41. Nepal, U.; Eslamiat, H. Comparing YOLOv3, YOLOv4 and YOLOv5 for Autonomous Landing Spot Detection in Faulty UAVs. *Sensors* **2022**, *22*, 464.
42. Ge, Z.; Liou, S.; Wang, F.; Li, Z.; Sun, J. YOLOX: Exceeding YOLO Series in 2021. *arXiv* **2021**, arXiv:2107.08430.
43. Asmara, R.A.; Syahputro, B.; Supriyanto, D.; Handayani, A.N. Prediction of Traffic Density Using Yolo Object Detection and Implemented in Raspberry Pi 3b + and Intel Ncs 2. In Proceedings of the 2020 4th International Conference on Vocational Education and Training (ICOVET), Malang, Indonesia, 19 September 2020; pp. 391–395. <https://doi.org/10.1109/ICOVET50258.2020.9230145>.
44. Md Abu Layek; Uddin, A.F.M.S.; Hossain, M.D.; Thu, N.T.; Yu, S.; Yong, C.H.; Lee, G.-W.; Chung, T.; Huh, E.-N. Cloud-Based Smart Surveillance System Using Raspberry Pi and YOLO. *Korea Softw. Congr.* **2018**, *12*, 510–512.
45. Wu, X.; Hong, D.; Ghamisi, P.; Li, W.; Tao, R. MsRi-CCF: Multi-Scale and Rotation-Insensitive Convolutional Channel Features for Geospatial Object Detection. *Remote Sens.* **2018**, *10*, 1990. <https://doi.org/10.3390/rs10121990>.
46. Gani, M.H.H.; Khalifa, O.O.; Gunawan, T.S.; Shamsan, E. Traffic Intensity Monitoring Using Multiple Object Detection with Traffic Surveillance Cameras. In Proceedings of the 2017 IEEE 4th International Conference on Smart Instrumentation, Measurement and Application (ICSIMA), Putrajaya, Malaysia, 28–30 November 2017; pp. 1–5. <https://doi.org/10.1109/ICSIMA.2017.8311983>.
47. Bommers, M.; Fazekas, A.; Volkenhoff, T.; Oeser, M. Video Based Intelligent Transportation Systems—State of the Art and Future Development. *Transp. Res. Procedia* **2016**, *14*, 4495–4504. <https://doi.org/10.1016/j.trpro.2016.05.372>.
48. Hoogendoorn, S.P.; Van Zuylen, H.J.; Schreuder, M.; Gorte, B.; Vosselman, G. Microscopic Traffic Data Collection by Remote Sensing. *Transp. Res. Rec.* **2003**, *1855*, 121–128. <https://doi.org/10.3141/1855-15>.
49. Toledo, T.; Koutsopoulos, H.; Ben-Akiva, M.; Jha, M. Microscopic Traffic Simulation: Models and Application. *Simul. Approaches Transp. Anal.* **2000**, *31*, 99–130. <https://doi.org/10.1007/0-387-24109-4>.
50. Chen, G.-W.; Yeh, T.-C.; Liu, C.-Y.; Ik, T.-U. Microscopic Traffic Monitoring and Data Collection Cloud Platform Based on Aerial Video. In Proceedings of the 2020 IEEE Wireless Communications and Networking Conference (WCNC), Seoul, Republic of Korea, 25–28 May 2020; pp. 1–6.
51. Castellano, G.; Castiello, C.; Mencar, C.; Vessio, G. Crowd Counting from Unmanned Aerial Vehicles with Fully-Convolutional Neural Networks. In Proceedings of the 2020 International Joint Conference on Neural Networks (IJCNN), Glasgow, UK, 19–24 July 2020; pp. 1–8.
52. Ciampi, L.; Amato, G.; Falchi, F.; Gennaro, C.; Rabitti, F. Counting Vehicles with Cameras. In Proceedings of the 26th Italian Symposium on Advanced Database Systems (SEBD 2018), Castellaneta Marina, Italy, 24–27 June 2018; Volume 2161.
53. Liu, Z.; Zhang, W.; Gao, X.; Meng, H.; Tan, X.; Zhu, X.; Xue, Z.; Ye, X.; Zhang, H.; Wen, S.; et al. Robust Movement-Specific Vehicle Counting at Crowded Intersections. In Proceedings of the 2020 IEEE/CVF Conference on Computer Vision and Pattern

- Recognition Workshops (CVPRW), Seattle, WA, USA, 14–19 June 2020; pp. 2617–2625. <https://doi.org/10.1109/CVPRW50498.2020.00315>.
54. Yang, H.; Zhang, Y.; Zhang, Y.; Meng, H.; Li, S.; Dai, X. A Fast Vehicle Counting and Traffic Volume Estimation Method Based on Convolutional Neural Network. *IEEE Access* **2021**, *9*, 150522–150531. <https://doi.org/10.1109/ACCESS.2021.3124675>.
  55. Patel, F.; Solanki, J.; Rajguru, V.; Saxena, A. Recognition of Vehicle Number Plate Using Image Processing Technique. *Adv. Emerg. Med.* **2018**, *7*, 2–8. <https://doi.org/10.18686/wct.v2i2.418>.
  56. Badr, A.; Abdelwahab, M.M.; Thabet, A.M.; Abdelsadek, A.M. Automatic Number Plate Recognition System. *Ann. Univ. Craiova Math. Comput. Sci. Ser.* **2011**, *38*, 62–71. <https://doi.org/10.52846/ami.v38i1.388>.
  57. Dalaff, C.; Reulke, R.; Kroen, A.; Ruhé, M.; Schischmanow, A.; Schlotzhauzer, G.; Tuchscherer, W.; Kahl, T. A Traffic Object Detection System for Road Traffic Measurement and Management. In Proceedings of the Image and Vision Computing New Zealand 2003, Palmerston North, New Zealand, 26–28 November 2003; pp. 78–83.
  58. Nasution, S.M.; Husni, E.; Kuspriyanto; Yusuf, R.; Mulyawan, R. Road Information Collector Using Smartphone for Measuring Road Width Based on Object and Lane Detection. *Int. J. Interact. Mob. Technol.* **2020**, *14*, 42–61.
  59. BPS Jumlah Penduduk Dan Keluarga Menurut Kecamatan Di Kota Bandung 2018 Dan 2019. Available online: <https://bandungkota.bps.go.id/statictable/2021/03/18/1437/jumlah-penduduk-dan-keluarga-menurut-kecamatan-di-kota-bandung-2018-dan-2019.html> (accessed on 26 July 2021).
  60. Mean Absolute Error. In *Encyclopedia of Machine Learning*; Sammut, C., Webb, G.I., Eds.; Springer: Boston, MA, USA, 2010; p. 652, ISBN 978-0-387-30164-8.

**Disclaimer/Publisher’s Note:** The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.