ISSN 1870-4069

Data Collection and Data Processing System for Traffic Studies

Hector Rodriguez-Rangel[1], Rafael Imperial Rojo[1],
Luis Alberto Morales Rosales[2], Sofia Isabel Fernandez Gregorio[3],
Abraham Efraim Rodríguez Mata[4], Peter Lepej [5]

[1] Tecnólogico Nacional de México,
Campus Culiacán,
Mexico

[2]Conacyt-Universidad Michoacana de San Nicolás de Hidalgo,
Mexico

[3] Instituto Tecnológico Superior de Martínez de la Torre,
Mexico

[4] Tecnólogico Nacional de México, Campus Chihuahua,
Mexico

[5]University of Maribor,
Slovenia

hector.rr@culiacan.tecnm.mx, imperial435@gmail.com,
imperial435@gmail.com, lamorales@conacyt.mx,
sfernandez@tecmartinez.edu.mx, abraham.rm@chihuahua.tecnm.mx,
peter.lepej@guest.um.si

Abstract. The usage of vehicles boosted urban mobility, which resulted in a rise in traffic accidents. Traffic studies (TS) using specialized equipment are necessary to investigate this phenomenon. It is now feasible to carry out these tasks without significantly altering the urban infrastructure because of technology advancements, artificial intelligence (AI), and films. The design of AI-based solutions requires generating public databases that provide reliable videos for the calibration and development of solutions that perform TS efficiently. This paper presents a system to generate a dataset from videos taken at an observation point on a roadway. The major highlight of these films is that they are not shot straight line with the camera. With the development of this system, we can assist in constructing various datasets to infer an object's speed and extract the attributes that the user believes to be the most essential. In addition, the system can detect, track, and offer object value information and combine it with their speed.

Keywords: Data base, vehicle speed, artificial vision.

1 Introduction

Automobiles are a necessity in our everyday life. More than 1.3 million people are killed in road accidents each year, according to the Association for Safe International Road Travel (ASIRT), with another 20 to 50 million wounded or handicapped ([15]). Traffic accidents are more common in the city center and suburbs than elsewhere, especially in Mexico. According to the National Institute of Statistics and Geographic Information (INEGI, 2011), accidents in the region grew by 72.7 percent in both urban and rural areas between 1997 and 2009 ([2]). The driver of an automobile carries the most responsibility in the case of an accident ([10]). Speeding, distracted driving, road obstructions, inadequate signaling, road infrastructure quality, and illumination are major contributors to traffic accidents. Because speeding is the most lethal element in road accidents, traffic studies are the primary source of information for identifying potential causes. These investigations necessitate the use of a reliable speed monitoring system. Furthermore, control systems such as Advanced Driver Assistance Systems (ADAS) help the driver while driving on the roadways [2].

The employment of special equipment has an impact on the usual technique of performing traffic studies. Traditional methods of collecting traffic data include those placed below the road surface (typically used for weigh-in-motion). For instance, inductive spires, magnetic field sensors (also above the road), galvanic contact devices/axle counters, capacitive and piezoelectric sensors (usually used for axle counting), or scales with sensor plates.

On the other hand, some sensors can be mounted above the road. For example, video surveillance cameras, microwave radar detectors, laser radar detectors, magnetic field sensors, passive and active infrared sensors, ultrasonic sensors, and other sound or video image processing instruments. A traffic study involves skilled experts and sensors at various places to properly analyze the sections that create congestion and/or accident concerns for examination inside cities and roads.

Video security cameras that provide real-time information are sensors that do not require any changes to the city's infrastructure. It is feasible to predict traffic flow parameters (e.g., vehicle density, peak hours, average speed, accident rate) by analyzing data from cameras, as well as identify congestion, accidents, and watch driver behavior ([12]). Besides, we can carry out processes such as traffic count, traffic modelling, traffic analysis, traffic prognosis, traffic safety studies (inspection and audit), and street network configuration from the video analysis. The main parameters are the speed and the trajectories of the vehicles. The traffic estimations can be provided to users and police patrols via road panels or integrated vehicle monitors to help in planning exits and avoiding traffic bottlenecks ([12]).

Artificial intelligence (AI) is changing the way we live in the modern world. Researchers and developers actively develop autonomous driving based on deep learning and monitoring techniques of the cars and road safety industries. However, before any artificial intelligence algorithm can be used in a production vehicle, it must first undergo a thorough functional safety evaluation and database construction. Therefore, we can see some advances and opportunities to solve problems of incorporating deep learning into autonomous cars in [6], as well as the construction of a database for training such neural networks.

This paper describes a technique for creating a dataset from movies collected at a roadside observation site. The major highlight of these films is that they are not shot perpendicular lines with the camera. With the development of this system, we can assist in constructing various datasets to infer object speed and extract the attributes that the user believes to be the most essential. Databases that gather diverse circumstances to monitor behaviors and calibrate their results are required to develop solutions for the autonomous creation of traffic studies. Besides, we want to remark that no credible public databases are available to assist with the Mexican traffic flow research.

2 Background

There are now several studies that calculate object speed; each one proposes a unique method, utilizing corpus such as the KITTI dataset [1]. This section highlights some of the works that were used as inspiration for this project.

Physics equations may be used with data from videos to estimate object's speed (1), acceleration (2), and angle (3) with a fast response. However, the straightforward application of these equations restricts their usage to a two-dimensional plane or the movement of vehicles parallel to the camera view.

$$Speed = \frac{Distance}{Time}, \tag{1}$$

$$Aceleration = \frac{Speed\,Final - Speed\,Initial}{Time\,Final - Time\,Initial}, \tag{2}$$

$$Angle = \tan^{-1} \frac{Point\,Final\,Y - Point\,Initial\,Y}{Point\,Final\,X - Point\,Initial\,X}. \tag{3}$$

In contrast, [8] proposes to determine the speed (equation 4), acceleration rate (equation 5) and the angle of an object (6)with data from a sequence of images with equation. These authors consider that the following characteristics must be satisfied:

- An approximate range of RGB values of the object is known.
- The rate at which the camera is taking images is known.
- The approximate size of the image is known.
- The image is represented in a uniform color.
- The background has uniform color and is not of the color of the object.

They use two images to calculate the acceleration of the vehicle. In the first and second images, a point is drawn at the centroid of the item to determine the desired values. The following formulae can be used to determine the object's speed, acceleration, and angle from both centroids:

$$Speed = \frac{\sqrt{(x2 - x1)^2 + (y2 - y1)^2,}}{FPS} \tag{4}$$

$$Acceleration = \frac{Speed\,Final - Speed\,Initial}{FPS}, \tag{5}$$

$$Angle = \tan^{-1}\frac{y2 - y1}{x2 - x1}. \tag{6}$$

In [4], authors use a multilayer perceptron neural network that was trained using picture sequences to detect a vehicle's relative speed, achieving good results with an average error of 1.12 m/s, which is equivalent to a LiDAR radar's error of 0.71 m/s. The vehicle track in a 2D plane, the depth, and optical flow estimations between successive pictures were recognized as three key elements for determining the speed. When recognizing a car in a succession of photos, we can see that its size changes based on its distance from the observer. This process impedes neural network learning; therefore, it was decided to divide the data into three categories: close, medium, and far, and train a model for each.

In [14], [13], authors determine the speed with the use of stereo cameras which are fixedly located in a strategic place in the city. The stereo cameras are responsible for identifying the license plate, headlights, and logo, as these characteristics are taken as main and always appear in a vehicle; these are used as a reference to determine the speed. In [9], a similar strategy is used by only identifying the license plate as the main character and using it as a reference to determine the speed of the vehicle. However, in this stage, only one camera is used instead of the stereo cameras used in [14, 13].

In [5] determine the speed of moving vehicles using drone-mounted stereo cameras. Their method integrates Mask-R-CNN and K-Means with the Lucas Kanade pyramid algorithm. In their experiments, they show that the combined use of Lucas Kanade, Mask-R-CNN and K-Means, improve their results compared to the use of these methods separately or the combined use of only 2 of them, even though they have performed experiments under normal circumstances, with many moving objects, and in low light circumstances.

In [3] develop a module for measuring the distance of vehicles in several lanes simultaneously using a drone. The drone is mounted with two LiDAR sensors, and each sensor emits a front point and a rear point, which vehicles are detected. The vehicle speed is estimated using the calculated distance between the front and rear point through which the vehicle passes and the time it takes to pass through both points.

3 Methodology

As stated in the introduction, this project aims to generate information on a road's traffic flow from video sequences. When conducting research, it was discovered that no dataset engaged the project's requirements. Thus

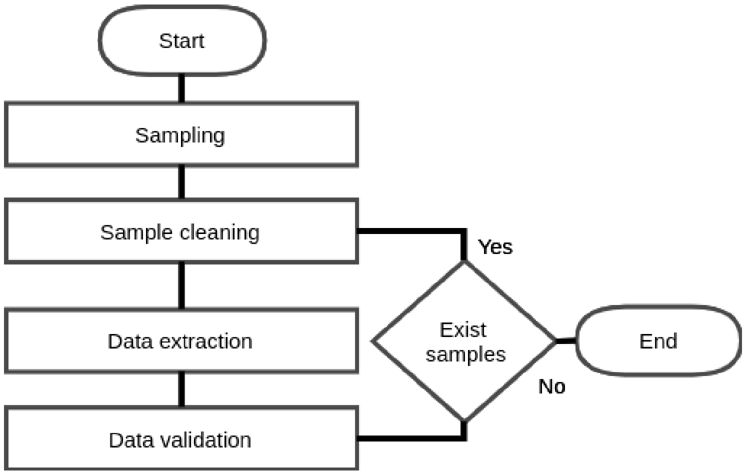

Fig. 1. Flowchart to generate data set.

it was essential to build one. This task will be discussed in this section of the methodology.

The procedure for extracting these features was broken down into four stages: 1) sampling, 2) sample cleaning, 3) data extraction, and 4) data validation. The whole procedure for creating the dataset is depicted in Figure 1.

3.1 Sampling

The first stage involves sampling, which required deploying two devices to retrieve the data set to construct the vehicle speed prediction model based on video sequences. The first one is a video camera located inside a smartphone. We configure the camera at 60 frames per second with Full HD resolution (1920 x 1080 pixels). We stabilize the camera so that the cars do not pass parallel to the camera view, preventing the computation in a 2D plane. The second one is a Bushnell radar (Figure 2) with a +/- 1.6 kilometer per hour accuracy to establish an accurate reference of the vehicle's speed analyzed to form the videos.

3.2 Sample Cleaning

The radar and the tracking system are not linked, then it is essential to combine the speed supplied by the radar with the information received by the system. For this, a visual inspection of the samples taken is required. The first step is to identify the Points in x that are used as a reference for taking them as vehicle entry and exit points. Using our Point of exit as a reference, we will identify the vehicles to which the speed was taken and write down the second it passes.

In addition, we identify the lane in which the vehicles pass to separate them in the three street lanes, we represent with zero the first lane, one the central

Fig. 2. Radar used for sampling.

Table 1. CSV output attributes

Attribute	Description
Output Angle	Angle from the input to the output.
Distance Traveled	Distance traveled in pixels from the input to the output.
Input Area	Area in pixels at the input.
Output Area	Area in pixels at the output.
FPS	Frames per seconds.
Time	Time to pass from the input to the output.
Lane	Street lane where pass the vehicle.
Speed	Speed detected by radar.
Identifier	Identifier to the created image.

lane, and two the last lane, with this we seek to create three datasets based on the work of [4].

This information obtained visually we save in a CSV file, which will have the following three attributes:

- The second the object passes through the exit point.
- The vehicle's speed.
- Lane.

3.3 Data extraction

When the term creation of the dataset is used, it refers to extracting the video's key features. These enable the determination of a vehicle's speed, frequency, and other critical parameters for investigating vehicular flow. The characteristics extracted are listed in Table 1, along with a brief explanation.

The execution of the system is simple; we need the CSV file generated earlier with the speeds and limits that we also identified in the previous step. The system

Fig. 3. Vehicle identified with its respective tracking.

is in charge of reading the video using the OpenCV library, and it examines frame by frame the content. Besides, it identifies the vehicles using the neural network YOLO ([7]), which identifies multiple objects in a single prediction; once it has identified all the vehicles, it draws the box corresponding to each one of them. This process allows the detection and identification of the objects of interest inside each frame.

The vehicles are tracked through a Kalman filter ([11]) to determine their location in the next frame. We choose this filter because it estimates a joint probability distribution over the variables for each timeframe. Besides, it has better performance when is used for linear or linearized processes and measurement systems than particle filter, which is more suitable for nonlinear systems. The system is in charge of preserving all the locations of the vehicles over time. Therefore, we can draw each vehicle's path inside the scenes and calculate the straight line corresponding to each trajectory. Figure 3 shows a detected vehicle in a white box, as well as a pair of lines, one yellow and another red, that correspond to the tracking of the same one and the calculated straight line corresponding to the tracking.

When the system detects a vehicle passing through the first boundary, it saves the frame to join it later, identifying when it passes through the exit point. It is worth noting that this is done for all detected vehicles, but only those that correspond to the second exit point identified in the CSV file are saved.

When the system detects a vehicle leaving the specified point, it records the generated data and speed in a line of a CSV file and an image of the vehicle entering and leaving the limits and an image identification. The technology operates at 30 to 40 frames per second, meaning each minute of video will take two minutes to process in the worst-case scenario.

Fig. 4. Valid system detection.

Fig. 5. Invalid system detection.

3.4 Data cleaning.

As previously stated, the system generates an image for each line of the resulting CSV. This image is an arrangement of two images: the image taken when the vehicle enters the entry point and the image taken when the vehicle exits the exit point; we use this image to verify that the vehicle was taken correctly by the system. In this situation, we consider legitimate pictures to be those that are not too far from the departure point and in which the vehicle has been identified in its whole or in a significant portion of it, as shown in Figure 4.

In the instance of incorrect images, we can observe how the system, in some stages, produces a partial output vehicle as a consequence. Figure 5 depicts an example where the system detects a cropped vehicle image; therefore, we delete it from the resulting CSV file. To remove invalid samples from the resulting CSV file, we must rely on the image identifier. This is found in the image name and is the last attribute of the CSV file. See Table 2.

The vehicle that the system is tracking to create its related information is shown in a white box in Figure 6, while the rest of the identified cars are shown in yellow and will not have a line generated for the resultant CSV.

4 Results

So far, implementing the system has generated a total of 178 videos ranging in length from one to two minutes, of which a total of 29 have been processed,

Fig. 6. Image generated by the system.

Fig. 7. Invalid image to the first lane.

from these 29 videos generated 532 samples, for each of the lanes the number of samples generated is as follows:

- First lane: 8 samples
- Middle lane: 239 samples
- Last lane: 285 samples

The first lane has few samples generated since this lane passes very close to the camera view, which causes a cut in the vehicle detection when passing to the exit point. Figure 7 shows one of these cases.

The following Table 2 shows us five random samples generated by the system, these samples were round to 3 decimals to be shown and the header description is in the Table 1.

We can use the video dataset to supply the system presented in previous sections, recognize, monitor, offer interest for traffic studies, and automatically calculate the vehicle's speed.

5 Conclusions

The system can provide a set of high-quality data. We may run tests with these data using various statistical and artificial intelligence approaches to forecast the speed at which objects occur in a video sequence. In addition, it allows us

Table 2. Example with data generated to the CSV.

Output Angle	Distance Traveled	Input Area	Output Area	FPS	Time	Speed	Lane	Identifier
80.458	714.973	26797	226200	58.063	0.878	45	1	246
79.570	879.673	25800	397794	58.063	0.964	66	1	407
75.300	778.597	25088	289962	60.018	0.600	61	0	6372
77.428	751.122	43960	368145	58.063	0.603	62	1	1380
82.959	718.533	9864	104535	58.063	1.240	57	2	1708

to combine sensors to create additional parameters that we can use to infer the object's speed. Also, it enables sensors integration to generate more parameters.

On the other hand, the generated dataset requires the removal of data that may affect the predictions. Hence, additional work is needed over the dataset by the possibles users. This process is a challenge to generate an acceptable dataset because of the time required to process all of the videos and validate the generated data.

A vital system improvement activity eliminates the need for a person to generate the CSV file with speed and validate that the information generated is correct. We aim to reduce the time required to generate the data set as future work related to the time spent by an operator because currently is relatively high, and the system could autonomously handle this task.

References

1. Burnett, K., Samavi, S., Waslander, S., Barfoot, T., Schoellig, A.: autotrack: A lightweight object detection and tracking system for the sae autodrive challenge, pp. 209–216 (2019) doi: 10.1109/CRV.2019.00036
2. Carro-Pérez, E. H., Ampudia-Rueda, A.: Conductas de riesgo al conducir un automóvil en zonas urbanas del sur de tamaulipas y la ciudad de méxico. CienciaUAT, vol. 13, no. 2, pp. 100–112 (2019)
3. Lee, K.-H.: A study on distance measurement module for driving vehicle velocity estimation in multi-lanes using drones. Applied Sciences, vol. 11, no. 9 (2021) doi: 10.3390/app11093884
4. Moritz Kampelmuhler, M. G. M., Feichtenhofer, C.: Camera-based vehicle velocity estimation from monocular video. Institute of Electrical Measurement and Measurement Signal Processing, (2018)
5. Peng, Y., Liu, X., Shen, C., Huang, H., Zhao, D., Cao, H., Guo, X.: An improved optical flow algorithm based on mask-r-cnn and k-means for velocity calculation. Multidisciplinary Digital Publishing Institute, (6 2019) doi: 10.3390/app9142808
6. Rao, Q., Frtunikj, J.: Deep learning for self-driving cars: Chances and challenges. In: Proceedings of the 1st International Workshop on Software Engineering for AI in Autonomous Systems. pp. 35–38 (2018)
7. Redmon, J., Divvala, S., Girshick, R., Farhadi, A.: You only look once: Unified, real-time object detection. In: 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). pp. 779–788 (2016) doi: 10.1109/CVPR.2016.91

8. Singh, S.: Estimating speed, velocity, acceleration and angle using image addition method. International Journal of Innovative Research in Computer and Communication Engineering, vol. 3, pp. 11585–11589 (11 2015) doi: 10.15680/IJIRCCE.2015
9. Vakili, E., Shoaran, M., Sarmadi, M.: Single-camera vehicle speed measurement using the geometry of the imaging system. Multimedia Tools and Applications, vol. 79 (07 2020) doi: 10.1007/s11042-020-08761-5
10. Velázquez Narváez, Y., Zamorano González, B., Ruíz Ramos, L.: Siniestralidad vial en la frontera norte de tamaulipas. enfoque en los procesos administrativos de control. Estudios fronterizos, vol. 18, no. 36, pp. 1–24 (2017)
11. Welch, G., Bishop, G., et al.: An introduction to the kalman filter (1995)
12. Yang, L., Li, M., Song, X., Xiong, Z., Hou, C., Qu, B.: Vehicle speed measurement based on binocular stereovision system. IEEE Access, vol. 7, pp. 106628–106641 (2019) doi: 10.1109/ACCESS.2019.2932120
13. Yang, L., Li, Q., Song, X., Cai, W., Hou, C., Xiong, Z.: An improved stereo matching algorithm for vehicle speed measurement system based on spatial and temporal image fusion. Entropy, vol. 23, no. 7 (2021) doi: 10.3390/e23070866
14. Yang, L., Luo, J., Song, X., Li, M., Wen, P., Xiong, Z.: Robust vehicle speed measurement based on feature information fusion for vehicle multi-characteristic detection. Entropy, vol. 23, pp. 910 (07 2021) doi: 10.3390/e23070910
15. Zaki, P. S., William, M. M., Soliman, B. K., Alexsan, K. G., Khalil, K., El-Moursy, M.: Traffic signs detection and recognition system using deep learning. arXiv preprint arXiv:2003.03256, (2020)