International Journal of Engineering Applied Sciences and Technology, 2022
Vol. 6, Issue 11, ISSN No. 2455-2143, Pages 190-194
Published Online March 2022 in IJEAST (http://www.ijeast.com)

REAL-TIME MOVING VEHICLE COUNTER SYSTEM USING OPENCV AND PYTHON

Preeti Bailke, Sanika Divekar
Department of IT
Vishwarkma Institue of Technology,
Pune, Maharashtra, India

***Abstract*— In view of highway management, intelligent vehicle recognition, and counting is becoming increasingly crucial. In highway monitoring, it is a crucial part that every vehicle should be observed and should go through vehicle monitoring and detection system. With today's technology have great effect on this area the road surveillance camera gives a large library of traffic video footage has been available for examination. By taking these video footage from the cameras the paper will present a system that will achieve to calculate the vehicle their direction with using image subtractor and counters. The computer vision will advance the capability of the system to process on the videos and give desired output. The main idea behind using computer vision and open CV is that they will not disturb the traffic and monitoring part will be done smoothly. The vehicle counter-offered in this study is primarily based totally on an aggregate of image processing algorithms which includes item detection, edges detection, and frames differentiation. The machine has been applied with the use of Python. This paper explains how to use multiple libraries for real-time image processing to achieve traffic flow counting and classification. The proposed system differs the existing system by providing the feature to count the cars in the day and night mode.**

***Keywords*— Traffic Monitoring, Computer Vision, Background subtraction, Contours, Video Detection**

I. INTRODUCTION

At first different Non-intrusive sensors were used where manual methods of video capturing used magnetic bar, radar, ultrasonic, acoustic, and video imaging sensors to detect and count vehicles. In today's advanced technology, intrusive sensors including such pneumatic road tubes, piezoelectric sensors, magnetic sensors, and an inductive loop are implemented for the same. Non-intrusive technology has benefits over intrusive technology, which necessitates the closure of travel lanes, endangering construction workers. Non-intrusive sensors were installed above the road surface and do not necessitate a traffic stop or lane restriction. Each sensor has its own benefits and drawbacks. However, manually counting by video or digital means will have high percentage of accuracy as compared to other technology.[1]
Because of the increase within side the variety of automobiles, expressways, highways, or even roads have become overcrowded. Counting the vehicle on road, tracking those moving cars, classification to the different categories, and Vehicle detection, are very essential in military, civilian, and authority's applications, which includes toll road monitoring, visitors planning, toll collection, and visitors flow. Vehicle detection is a vital step in highway traffic management. This method presents an affordable, portable, and Computer Vision-based detection and counting system moving vehicles. Images from a video sequence are converted into frames to detect moving vehicles after extracting the background. The system is built with OpenCV by using its provided implementing kits and other libraries with the help of real time videos which are stored from the planted cameras on the highway. The system which are built to count the moving vehicles with the help of following methods background subtraction, image filtering, converting image to grey, and MOG algorithm that is Gaussian Mixture-primarily based totally Background/Foreground Segmentation Algorithm. This system can also count passing vehicles from pre-recorded video utilizing CV techniques including thresholding, hole-filling, and adaptive morphological dilation to remove the noise and emphasize the foreground item. [5]
Many features such as lighting, the shadow of the object, border of the vehicle can be used during the building of vehicle counting systems. Vehicle detection is crucial part to implement through the technology even with advance in the technology there are parameter which can affect the output of system one of the parameter is the correct brightness and light in the video frames, the brightness will be the main source to count the vehicle. In the proposed system, the video of any traffic monitoring system can be set as an input which will follow the different stages, the first part of the system explain how the video fetched from the database should be convert into individual frames which will be an array of that, the second part of the system will extract the backgrounds by subtracting the vehicle from the scenery, and in the last stage system will moving vehicle detection with the direction and also present the count with using OpenCV and built in python.

International Journal of Engineering Applied Sciences and Technology, 2022
Vol. 6, Issue 11, ISSN No. 2455-2143, Pages 190-194
Published Online March 2022 in IJEAST (http://www.ijeast.com)

II. BACKGROUND

A. Video Processing –

Digital Signal Processing (DSP) includes video processing as a subcategory that takes digitized real-world signals such as voice, audio, video or position, and alters them mathematically. Using image processing, frames can be obtained from a video and converted into images where each individual image is called a frame per second (FPS). Motion is simply realized in this situation by comparing subsequent frames. The OpenCV for Video processing can be used to highlight certain regions of videos, remove improper lighting effects, eliminate camera motions, and remove edge-artifacts with cv2.resize is an example for resizing image [1]. Python's OpenCV library provides a number of features. Numpy is used for creating vectors and matrices which store information about the video after converting it into frames. OpenCVPython, a package with MATLAB-style syntax, is used for numerical computations on the frames. The frames extracted from video sequence are all array structures which are transformed using Numpy. [2].

B. RGB to Greyscale Conversion -

Image processing using OpenCV is used in video analysis but in the first preprocessing part it is mandatory to convert the RGB format image into the grey scale image which will give more accuracy. The major purpose of this conversion is to produce more acceptable outcomes while processing grayscale photos than processing RGB ones [5]. As discussed previously the conversion of RBG to grey scale is very essential part for the system working there is RGB color mode which will be transformed into the grey scale values. The grey scale values are basically denoted by 0 to 255. This will enhance the video processing technique for the system. When changing an RGB image to grayscale mode, certain values are presented as the outcome the main parameter will be the percentage of the brightness pixel which the images contains. This brightness pixel will be the value as an outcome.

III. PREVIOUS WORK

Several techniques to building systems which can recognize count, and categorize vehicles for use in a suggested surveillance model in an Intelligent Transportation System have been examined. This chapter covered both a study of variety system which implemented this type of project and the variety of techniques which helped to build these system.

The system which caught attention from the different researcher have the main common field is video/image processing. This area got attention in the recent years. One of the following methodologies has been used to detect and track vehicle detection [8]:

- Matching
- Threshold and segmentation
- Point detection
- Edge detection
- Frame differentiation

Describes one of the most essential object detection studies that helped in the development of auto scope video detection systems. The system also presents how the moving vehicles are differentiated from the highway road video sequence; forward and backward image differencing algorithms [10] are applied. Feature vectors from image regions [4], [9] prove to be quite effective for vehicle detection. Accurate vehicle dimension estimation using a set of coordinate mapping functions [7] is depicted. A variety of boosting algorithms for object detection [8], [5] utilizing machine learning approaches that can recognize and classify moving objects is implemented.

An optimized virtual loop approach [12] for a video-based real-time vehicle counting system has been proposed. In this approach, real-time traffic monitoring cameras are installed on roads to calculate the number of vehicles passing through. Counting is accomplished using three steps namely: method of employing backdrop subtraction, method of using continuous video frame difference, and counting the vehicle from the Region of Interest (ROI) zone.

Lei, M., et al. [13] proposed yet another video-based vehicle counting technique. Surveillance cameras were employed in this system and were positioned at a somewhat high elevation to gather the traffic video stream. Two main approaches are used in this system namely Gaussian shadow eradication and adaptive background detection The 1.2 RGB To Greyscale Conversion viewing angle, as well as the system's capacity to eliminate shades and illusions, determine the system's accuracy. In video analysis, OpenCV image processing is utilized to create effects. The system's fundamental problem is its failure to distinguish vehicle types.

IV. PROPOSED SYSTEM

The proposed system consists of three stages:

1. System Initialization: In this stage, the system is set up and initialized. A constant stream of video is recorded by the camera and manually the video stream is transferred for processing.
2. Background Subtraction: This is important stage where the system take the frames from video sequence, and background subtraction is performed on the frames that have been received from the video using OpenCV.
3. Vehicle Detection: At this stage, the moving vehicle which are extracted from the previous stage are counted using the removed background image. The proposed system is suitable for the prerecorded videos from the database.

International Journal of Engineering Applied Sciences and Technology, 2022
Vol. 6, Issue 11, ISSN No. 2455-2143, Pages 190-194
Published Online March 2022 in IJEAST (http://www.ijeast.com)

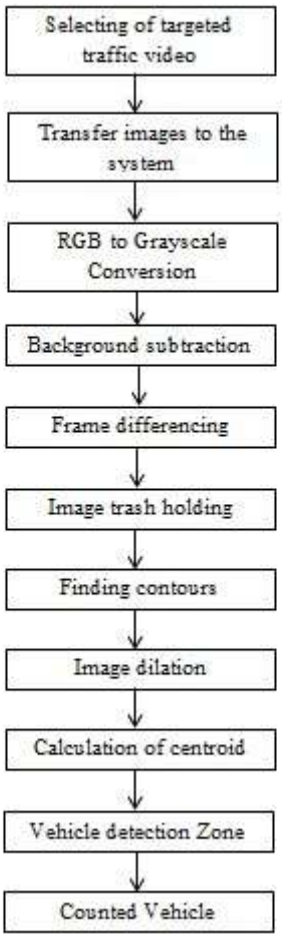

Fig.1. The Architecture of The Proposed System

4.1 Grayscale Image Generation and Image Enhancement:
The first stage of the system explains in the block diagram is that the frame/images taken from video must be converted into grey scale format for the better accuracy. Hence, each video frame undergoes RGB to grayscale conversion.
To achieve an appropriate threshold value, each frame should be placed in contrast to the foreground, and then the input image should be transformed.. The power-law approach was utilized in this study as one of several grayscale conversions. The function cv2.cvtColor (input image, flag) for color conversion is used, where flag indicates the type of conversion. The parameter cv2.COLOR BGR2GRAY is used to convert to grayscale.[1]

Fig.2. Input to the system as RGB format

Fig.3. The RGB video converted into greyscale format

4.2 Edge Detection:
Edge detection is an image processing approach that identifies points in a digital image that have discontinuities, or sudden changes in image brightness. The borders (or boundaries) of an image are the points where the image brightness fluctuates drastically. For the edge each picture (video frame) comprises three key features viz. point, edge, and curve. Among the attributes (point, edge, and curve) stated, edge pixels are a good choice. Edge pixels, which are the major features of passing automobiles in a roadway video frame, can be detected by processing picture pixels [2].
The canny operator, which has been employed in this study, is one of the most used approaches to detect the edges of an image. Figure 4 shows the original image, while Figure 5 shows image after conversion using morphological transformations. Morphological transformations are a set of straightforward procedures depending on shape of a picture. It's usually done with binary images. It requires two inputs: one is our original image, and the other is a structuring element or kernel that determines the operation's nature. Erosion and Dilation are simple and basic morphological operators. Then there are variations such as Opening, Closing, and Gradient etc. The output of the edge detection procedure is shown in a fig5 of the identified edge pixels, as can be seen.

International Journal of Engineering Applied Sciences and Technology, 2022
Vol. 6, Issue 11, ISSN No. 2455-2143, Pages 190-194
Published Online March 2022 in IJEAST (http://www.ijeast.com)


Fig.4. Original Image

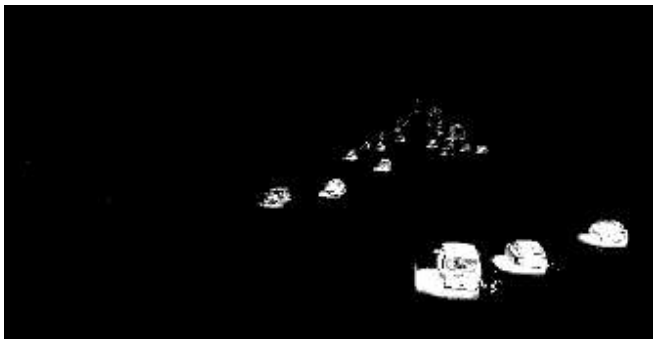
Fig.5. Edge Detection Result

4.3 Background Subtraction Algorithm:
BackgroundSubtractorMOG2 is the algorithm that was used to implement the proposed system. Background Subtraction is a Gaussian Mixture-based Background/Foreground Segmentation Algorithm. [14] It employs an automatic way to select a suitable number of Gaussian mixtures for the pixel, unlike [5], which identifies the amount of regions to be used in the model's development. Furthermore, the algorithm is better at dealing with changes in lighting in the scene. The algorithm also allows you to specify whether or not the shadows of the objects should be detected.
Shadows will be detected and marked by the algorithm. For cases like this, several methods have been presented; for the proposed system, Background Subtraction MOG algorithm is used from the library OpenCV[4], which uses Gaussian distributions to generate a model of the image's back dropping. MOG employs a technique that involves combining K Gaussian distributions to describe each backdrop pixel (K is 3 to 5). Background Subtractor GMG is a background subtraction implementation in OpenCV that is based on images [11] and incorporates both background image estimation and Bayesian segmentation.

4.4 Contour Extraction:
Contours are the shape's guidelines that are used in curve recognition and detection. The precision with which the process of contour detection using canny edge detection is carried out on a binary image. The cv2.findContours() technique in the OpenCV library is being used to find contours. [3]

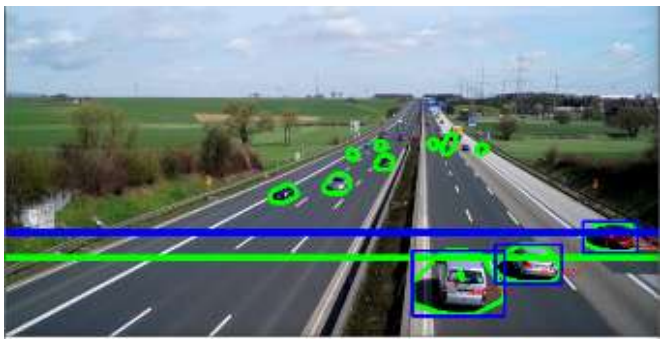
Fig.6. Green Contour of Vehicle

4.5 Counting Vehicle:
To count the moving vehicle the lines present in figure 6 will perform the crucial part. The colored lines as blue and green will the direction of the moving vehicle which identifies the Region of Interest (ROI).[3] As ROI along with the imaginary line plays an important role in categorization, ROI requires close human monitoring [9]. For ROI, the system executes the following sequence of operations: applying a background mask, deleting the mask, executing binary threshold, morphology with erosion and dilation, and converting the frame to grayscale. Following these actions, contours are discovered [4].
For the moving vehicle part to count the vehicle the system have implemented the centroid which will have condition as set for a vehicle has tried to touch or managed to cross an imaginary line in ROI. When 2 ROI coordinates are connected diagonal direction, a line called the imaginary line appears. The vehicle is counted once the vehicle manages to cross centroid in ROI which is imaginary line. From fig. 6, which have different parameter in the box as cars in area the area is names as ROI, the cars crossed up/down means direction of moving cars fetched through the centroid. The car will pass from the centroids and will tell the direction and after that the system will also count the number of moving vehicles. The vehicles which are passing through the ROI will be counted and then total counter will be update.

Fig.7. When the vehicle centroid crosses the imaginary line in ROI, the respective counter variables increase.

V. EXPERIMENT AND RESULT

To implement this project the system used real time data as for input in the form of videos divided into frames where vehicle detection , counting, vehicle direction, and the counting of vehicle is done with the proposed techniques.
The proposed system uses OpenCV with very straightforward way for ordinary vehicle detection and counting of the cars. Applying in Computer Vision and, in particular, object counting to vehicle detection and counting overcomes these drawbacks in the blink of an eye – no pun intended. The computer vision technology allows measuring traffic flow from the standard CCTV camera stream making our solution non-intrusive, fully wireless and easy to install or adjust.
The experimental outputs have proved that proposed real-time approach achieves high accuracy for counting vehicles. Experiments on different real videos from any dataset show that results are excellent and proposed approach outperformed with output..

VI. CONCLUSION

The proposed system suggested a method for detecting and tracking objects in highway surveillance video and established vehicle detection from the standpoint of surveillance cameras. NumPy, matplotlib, and scipy are examples of helpful Python libraries that may be used to count vehicles and classify traffic. Compared to typical hardware-based traffic monitoring, the method described in this study is low-cost and stable, and it does not necessitate large-scale construction or installation work on existing monitoring gadgets. The system is also able to detect vehicles at night time. The system requires the presence of foreground objects in order to collect contour properties and features.

VII. REFRENCES

[1] Reha Justin, Dr. Ravindra Kumar, "Vehicle Detection and Counting Method Based on Digital Image Processing in Python," International Journal of Electrical Electronics & Computer Science Engineering -2018.

[2] Nisar, M. A., Kumari, A., Jethwa, H., & Chasker, N, "Real-time traffic light control using image processing" International Journal for Scientific Research and Development 2016, April 1.

[3] Sania Bhatti, Mir Muhammad B. Talpur, & Mohsin A. Memon, "A Video based Vehicle Detection, Counting and Classification System" http://www.mecs-press.org, DOI: 10.5815, 2018, September 8.

[4] P. KaewTraKulPong, and R. Bowden, "An improved adaptive background mixture model for real-time tracking with shadow detection", Video-based surveillance systems, Springer. pp. 135-144, 2002.

[5] N. Seenouvong, U. Watchareeruetai, C. Nuthong, K. Khongsomboon, "A Computer Vision Based Vehicle Detection and Counting System", IEEE 8th International Conferenceon Knowledge and Smart Technology (KST), pp.224-227, 2016

[6] P. Choudekar, S. Banerjee, M. K. Muju, "Real-time Traffic Light Control Using Image Processing," Indian Journal of Computer Science and Engineering, Vol. 2, No. 1, ISSN: 0976-5166I. pp. 271–350.

[7] R. Gonzalez, R. E. Woods, "Digital Image Processing," 2nd Edition, Prentice-Hall, 2002.

[8] N. Chintalacheruvu, V. Muthukumar, "Video Based Vehicle Detection and Its Application in Intelligent Transportation Systems," Journal of Transportation Technologies, Vol. 2, pp. 305-314, 2012.

[9] P. Choudekar, S. Banerjee, M. K. Muju, "Real-time Traffic Light Control Using Image Processing," Indian Journal of Computer Science and Engineering, Vol. 2, No. 1, ISSN: 0976-5166.

[10] D. G. Lowe, "Distinctive Image Features from Scale-Invariant Keypoints," International Journal of Computer Vision, pp. 91-110, 2004.

[11] M. Tursun, and G. Amrulla, "A video based real-time vehicle counting system using optimized virtual loop method", IEEE 8th International workshop on Systems Signal Processing and their Applications (WoSSPA), 2013.

[12] M. Lei, D. Lefloch, P. Gouton, K. Madani, "A video based real-time vehicle counting system using adaptive background method", IEEE International Conferenceon Signal Image Technology and Internet Based Systems (SITIS'08), pp. 523-528, 2008.

[13] Sania Bhatt, Liaquat A. Thebo, Mir Muhammad B. Talpur,Mohsin A. Memon,"A Video based Vehicle Detection, Counting and Classification System", MECS I.J. Image, Graphics and Signal Processing,September 2018.

[14] Z. Zivkovic, "Improved adaptive Gaussian mixture model for background subtraction", 17th International Conferenceon Pattern Recognition(ICPR), 2004.